# EPD®.Visualizer User Guide

Optegra® Release 6

DOC36761-011

Parametric Technology Corporation

**Copyright © 2001 Parametric Technology Corporation. All Rights Reserved.**
User documentation from Parametric Technology Corporation (PTC) is subject to copyright laws of the United States and other countries and is provided under a license agreement, which restricts copying, disclosure, and use of such documentation. PTC hereby grants to the licensed user the right to make copies in printed form of PTC user documentation provided on software or documentation media, but only for internal, noncommercial use by the licensed user in accordance with the license agreement under which the applicable software and documentation are licensed. Any copy made hereunder shall include the Parametric Technology Corporation copyright notice and any other proprietary notice provided by PTC. User documentation may not be disclosed, transferred, or modified without the prior written consent of PTC and no authorization is granted to make copies for such purposes.

Information described in this document is furnished for general information only, is subject to change without notice, and should not be construed as a warranty or commitment by PTC. PTC assumes no responsibility or liability for any errors or inaccuracies that may appear in this document.

The software described in this document is provided under written license agreement, contains valuable trade secrets and proprietary information, and is protected by the copyright laws of the United States and other countries. UNAUTHORIZED USE OF SOFTWARE OR ITS DOCUMENTATION CAN RESULT IN CIVIL DAMAGES AND CRIMINAL PROSECUTION.

**Registered Trademarks of Parametric Technology Corporation or a Subsidiary**
Advanced Surface Design, CADDS, CADDShade, Computervision, Computervision Services, Electronic Product Definition, EPD, HARNESSDESIGN, Info*Engine, InPart, MEDUSA, Optegra, Parametric Technology, Parametric Technology Corporation, Pro/ENGINEER, Pro/HELP, Pro/INTRALINK, Pro/MECHANICA, Pro/TOOLKIT, PTC, PT/Products, Windchill, InPart logo, and PTC logo.

**Trademarks of Parametric Technology Corporation or a Subsidiary**
3DPAINT, Associative Topology Bus, Behavioral Modeler, BOMBOT, CDRS, CounterPart, CV, CVact, CVaec, CVdesign, CV-DORS, CVMAC, CVNC, CVToolmaker, DesignSuite, DIMENSION III, DIVISION, DVSAFEWORK, DVS, e-Series, EDE, e/ENGINEER, Electrical Design Entry, Expert Machinist, Expert Toolmaker, Flexible Engineering, *i*-Series, ICEM, Import Data Doctor, Information for Innovation, ISSM, MEDEA, ModelCHECK, NC Builder, Nitidus, PARTBOT, PartSpeak, Pro/ANIMATE, Pro/ASSEMBLY, Pro/CABLING, Pro/CASTING, Pro/CDT, Pro/CMM, Pro/COMPOSITE, Pro/CONVERT, Pro/DATA for PDGS, Pro/DESIGNER, Pro/DESKTOP, Pro/DETAIL, Pro/DIAGRAM, Pro/DIEFACE, Pro/DRAW, Pro/ECAD, Pro/ENGINE, Pro/FEATURE, Pro/FEM-POST, Pro/FLY-THROUGH, Pro/HARNESS-MFG, Pro/INTERFACE for CADDS 5, Pro/INTERFACE for CATIA, Pro/LANGUAGE, Pro/LEGACY, Pro/LIBRARYACCESS, Pro/MESH, Pro/Model.View, Pro/MOLDESIGN, Pro/NC-ADVANCED, Pro/NC-CHECK, Pro/NC-MILL, Pro/NC-SHEETMETAL, Pro/NC-TURN, Pro/NC-WEDM, Pro/NC-Wire EDM, Pro/NCPOST, Pro/NETWORK ANIMATOR, Pro/NOTEBOOK, Pro/PDM, Pro/PHOTORENDER, Pro/PHOTORENDER TEXTURE LIBRARY, Pro/PIPING, Pro/PLASTIC ADVISOR, Pro/PLOT, Pro/POWER DESIGN, Pro/PROCESS, Pro/REPORT, Pro/REVIEW, Pro/SCAN-TOOLS, Pro/SHEETMETAL, Pro/SURFACE, Pro/VERIFY, Pro/Web.Link, Pro/Web.Publish, Pro/WELDING, Product Structure Navigator, PTC *i*-Series, Shaping Innovation, Shrinkwrap, The Product Development Company, Virtual Design Environment, Windchill e-Series, CV-Computervision logo, DIVISION logo, and ICEM logo.

**Third-Party Trademarks**
Oracle is a registered trademark of Oracle Corporation. Windows and Windows NT are registered trademarks of Microsoft Corporation. Java and all Java based marks are trademarks or registered trademarks of Sun Microsystems, Inc. CATIA is a registered trademark of Dassault Systems. PDGS is a registered trademark of Ford Motor Company. SAP and R/3 are registered trademarks of SAP AG Germany. FLEX*lm* is a registered trademark of GLOBEtrotter Software, Inc. VisTools library is copyrighted software of Visual Kinematics, Inc. (VKI) containing confidential trade secret information belonging to VKI. HOOPS graphics system is a proprietary software product of, and copyrighted by, Tech Soft America, Inc. All other brand or product names are trademarks or registered trademarks of their respective holders.

UNITED STATES GOVERNMENT RESTRICTED RIGHTS LEGEND
This document and the software described herein are Commercial Computer Documentation and Software, pursuant to FAR 12.212(a)-(b) or DFARS 227.7202-1(a) and 227.7202-3(a), and are provided to the Government under a limited commercial license only. For procurements predating the above clauses, use, duplication, or disclosure by the Government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software Clause at DFARS 252.227-7013 or Commercial Computer Software-Restricted Rights at FAR 52.227-19, as applicable.

**Parametric Technology Corporation, 140 Kendrick Street, Needham, MA 02494-2714**
**8 January 2001**

# Table of Contents

## Preface


Related Documents ____ xxi
Book Conventions ____ xxii
Online User Documentation ____ xxiii
Printing Documentation ____ xxiii
Resources and Services ____ xxiv
Documentation Comments ____ xxiv


## Introduction to EPD.Visualizer


What Is EPD.Visualizer? ____ 1-2
How Can You Use EPD.Visualizer? ____ 1-3
EPD.Visualizer Features ____ 1-3
Advanced Rendering Module ____ 1-4
Basic Concepts ____ 1-5
System Overview ____ 1-5
Objects and Items ____ 1-6
Neutral Graphic Files ____ 1-7
Metafiles ____ 1-7
3D Viewing ____ 1-8
Coordinate System ____ 1-8
Viewing ____ 1-9
Linear and Angular Movement ____ 1-9

Using the EPD.Visualizer Window ____ 1-10
Graphics Window ____ 1-11
Prompt and Status Windows ____ 1-11
Palettes ____ 1-11
Tool Palette and Other Palettes ____ 1-12
Slider Scales ____ 1-13
Text Editor ____ 1-14
Menu Bar ____ 1-14
Dialog Boxes ____ 1-15
Text Box ____ 1-15
Lists ____ 1-15
Option Buttons ____ 1-16
Tools ____ 1-16
Buttons ____ 1-16
File Selection Dialog Box ____ 1-17
Slider Scale ____ 1-18
Color Editor ____ 1-19
Defining a New Color ____ 1-19
Deleting a Color ____ 1-20
Restoring a Deleted Color ____ 1-20
Selecting or Verifying a Color ____ 1-20
Permanent Mouse Dynamics ____ 1-21
Screen Axis Motions ____ 1-21
Behavior of Screen Axis Motions ____ 1-22
Example of Screen Axis Motions ____ 1-23
Model Axis Motions ____ 1-26
Behavior of Model Axis Motions ____ 1-26
Example of Model Axis Motions ____ 1-27
Quick View Centering ____ 1-28
Considerations ____ 1-28
Starting an EPD.Visualizer Session ____ 1-29
Running EPD.Visualizer ____ 1-29
Adding Graphic Files or Opening Metafiles ____ 1-29
Starting a New Session ____ 1-29
Obtaining Session Information ____ 1-30
Quitting EPD.Visualizer ____ 1-30

# Basic Features


Display Controls ____ 2-2
Rendering ____ 2-2
Perspective Angle ____ 2-3

Light Management ____ 2-5
Directional Lighting ____ 2-5
Point Lighting ____ 2-5
Spot Lighting ____ 2-5
Matrix Duplication of Lights ____ 2-5
Ambient Lighting ____ 2-6
Environmental Brightness ____ 2-6

View Processing ____ 2-7
Showing and Setting the Primary View ____ 2-7
Viewing All Items ____ 2-7
Saving Views and Notes ____ 2-7
Defining and Playing Back View Sequences ____ 2-7

Selecting Items ____ 2-9
Building a Selection List ____ 2-9
Setting Selection Masks ____ 2-9
Selecting and Deselecting Items ____ 2-9
Defining Selection Sets ____ 2-10

Verification ____ 2-11
Verify ____ 2-11
Measure Distance from Camera to Item ____ 2-11
Measure Distance Between Items or Graphics Files ____ 2-12
Measuring Distance in Sections ____ 2-12

Item Editing ____ 2-13
Controlling Visibility ____ 2-13
Deleting Items ____ 2-13
Translating and Rotating Items ____ 2-14
Spatial Orientation ____ 2-14
Target of Translation and Rotation Operations ____ 2-15

Translation ____ 2-15
Rotation ____ 2-16
Item Traversal ____ 2-16
Dynamic Interference Checking ____ 2-17
Item Color ____ 2-17
Surface Materials ____ 2-17
Traversal ____ 2-19
Defining Path and Gpaths ____ 2-19
Camera Traversal ____ 2-19
Item and Section Plane Traversal ____ 2-20
Interference Checking ____ 2-21
Static Interference Checking ____ 2-21
Dynamic Interference Checking ____ 2-21
Text ____ 2-22
Defining Text Properties ____ 2-22
Creating, Editing, and Deleting Text ____ 2-22
Attaching Text or Tags to Objects ____ 2-22
Accessing Online Help ____ 2-23

# Using EPD.Visualizer Menus

EPD.Visualizer Main Menus ____ 3-2
File Menu ____ 3-3
Starting a New Session ____ 3-3
Opening a Metafile ____ 3-4
Adding Textual or Binary Graphics Files ____ 3-5
Removing Graphic Files ____ 3-7
Saving a Metafile ____ 3-8
Graphic Output ____ 3-9
Capturing a Raster Image ____ 3-9
PostScript Graphics Output ____ 3-11
Obtaining Session Information ____ 3-12
Quitting EPD.Visualizer ____ 3-13

Edit Menu 3-14
Show 3-14
Hide 3-14
Show All 3-15
Hide All 3-15
Item Dynamics 3-15
Path Traversal 3-16
Translate/Rotate 3-22
Restoring Positions After Translation or Rotation 3-27
Path Editor 3-28
The Path Editor 3-30
The Path Router 3-31
Visual Properties 3-34
Item Color 3-34
Surface Materials 3-35
Text Properties 3-38
Delete 3-40
View Menu 3-41
Primary View 3-41
Show 3-41
Set 3-41
View 3-42
All 3-42
Visible 3-42
Changing Views 3-43
Secondary Views 3-44
Perspective 3-45
Rendering 3-45
Shadowing Controls 3-48
Ray Trace Controls 3-48
Radiosity Controls 3-50
Lighting 3-51
Editor 3-51
Environment 3-60
View Editor 3-60

View Sequence ____ 3-62
Editor ____ 3-62
Playback ____ 3-64
Camera Traversal ____ 3-66
Before You Start ____ 3-66
Start Camera Traversal ____ 3-69
Verify the Path Name ____ 3-69
Blend Colors ____ 3-69
Movie Recorder ____ 3-70
Options Menu ____ 3-73
Hide/Show Palette ____ 3-73
Left/Right Palette ____ 3-73
Preferences ____ 3-74
Temp Color ____ 3-75
Tools Menu ____ 3-76
Tree ____ 3-76
Item ____ 3-77
Mouse Dynamics ____ 3-77
Zoom ____ 3-77
Set Camera ____ 3-77
Measure Distance ____ 3-78
Zone Definition ____ 3-78
Normals ____ 3-78
Select Menu ____ 3-79
Selection Masks ____ 3-79
Selection Set Editor ____ 3-80
Build Selection List ____ 3-81
Copy Selection to Secondary Selection ____ 3-82
Deselect All ____ 3-82
Clear Secondary Selection ____ 3-83
Highlight Selection ____ 3-83

Database Menu 3-84
Connect 3-84
ASCII 3-84
DBMS 3-86
Disconnect 3-86
Query Types 3-86
Selection Keys 3-88
Substitution Variables 3-88
Query Database 3-90
Select a Query from a File 3-91
Select a Query From DBMS 3-92
Query Database Data 3-93
Save a DBMS Report 3-94
Manual or Automatic Tagging 3-95
Interference Checking Menu 3-96
Sectioning Menu 3-97
Online Help Menu 3-98
On Help 3-98
On Context 3-98
On General 3-98
On Glossary 3-99
On Environment 3-99
On Release Notes 3-99

# Constructing Zones

About Zones 4-2
Environment Variables You Need to Set 4-3
Creating Your Zone Definition 4-4
Creating a Box Zone 4-4
Corners Box Type 4-6
Radius Box Type 4-6
Selection List 4-7
Creating a Cylinder Zone 4-7

# Interference Checking


About Interference Checking _____ 5-2

Starting Interference Checking _____ 5-3

Setting Up an Interference Checking Session _____ 5-4

- Turning On or Off Item Dynamics _____ 5-4
- Coloring and Highlighting Colliding Components _____ 5-5
  - Applying Color to Components _____ 5-5
  - Viewing Results Without Color _____ 5-5
  - Displaying the Selected Object in Blue _____ 5-5
  - Making Selected Objects Transparent _____ 5-5
  - Highlighting Colliding Components in EPD.Connect _____ 5-6
- Checking Against the Selection _____ 5-6
  - Checking Options _____ 5-6
- Creating Secondary Selections _____ 5-7
- Setting Analysis Options _____ 5-7

Reviewing the Results _____ 5-9

- Viewing the Interferences _____ 5-9
- Obtaining a Report _____ 5-10
- Report Content _____ 5-10
- Filing the Report _____ 5-11


# Sectioning


About Sectioning _____ 6-2

Starting Your Sectioning Session _____ 6-3

Displaying the Main Sectioning Window _____ 6-4

Defining a Plane _____ 6-5

- Positioning the Plane _____ 6-5
- Orienting the Plane _____ 6-6
- Defining by Points _____ 6-6

# Controlling a Plane's Display and Multiple Sectioning


Getting Started ____ 7-2

Displaying the Plane ____ 7-3
- Specifying the Size ____ 7-3
- Setting the Display Characteristics ____ 7-4
- Defining the Color ____ 7-4

Clipping the Scene ____ 7-5
- Specifying the Thickness ____ 7-5
- Reversing the Normal Vector ____ 7-5
- Viewing the Plane ____ 7-5

Sectioning the Scene ____ 7-6
- Displaying the Section in a Separate View ____ 7-6
  - Mouse Behavior in the Section Window ____ 7-8
  - View Options ____ 7-9
- Interference Checking ____ 7-11
  - Setting a Proximity Tolerance ____ 7-12
  - Setting a Contact Distance ____ 7-13
- Selecting the Objects To Be Sectioned ____ 7-13

Sectioning the Scene from Multiple References ____ 7-14
- Configuring Multiple Sectioning ____ 7-17
- Performing Multiple Sectioning ____ 7-19
- Saving Multiple Sectioning ____ 7-20

Measuring Distance in Sections ____ 7-20
- Distance Between Two Points ____ 7-21
- Distance Between Two Endpoints ____ 7-23
- Minimum Distance Between a Point and a Section ____ 7-24
- Minimum Distance Between an Endpoint and a Section ____ 7-25
- Minimum Distance Between Two Sections ____ 7-26
  - Considerations ____ 7-27

Verifying a Point ____ 7-28

Saving a Section ____ 7-29
- Save Options ____ 7-30

Output File Format _____ 7-32
Description of a Section _____ 7-32
Description of a Sectioned Graphic File_____ 7-33
Listing Interferences _____ 7-34
Example File _____ 7-34
Section Window Graphics Output _____ 7-36

# Moving a Plane

Getting Started _____ 8-2
Setting Plane Dynamics_____ 8-3
Moving the Plane Without Objects Selected _____ 8-3
Orienting_____ 8-4
Translating and Rotating _____ 8-5
Moving the Plane with Objects Selected_____ 8-6
Translating the Plane _____ 8-7
Moving by Steps _____ 8-7
Moving by Digitizing _____ 8-7
Rotating the Plane _____ 8-8
Moving the Plane to a Reference _____ 8-9
Moving the Plane to the Selection List _____ 8-10
Setting and Retrieving the Primary Plane _____ 8-11

# Palettes

Tool Palette _____ 9-2
Select _____ 9-3
Volume Select _____ 9-4
Verify _____ 9-4
Tree Select and Scroll_____ 9-5
Permanent Zoom_____ 9-5
Zoom _____ 9-6
Set Camera Position _____ 9-6
Set Camera Target _____ 9-7

Set Camera Position and Target ____ 9-7
Measure Distance from Camera to Item ____ 9-8
Measure Distance Between Items ____ 9-8
Mouse Viewing ____ 9-9
Deselect ____ 9-12
Text ____ 9-12
Object Tagging ____ 9-13
Manually Tag Objects ____ 9-13
Automatically Tag Objects ____ 9-13
Walk-Through Palettes ____ 9-15
Move Palette ____ 9-16
Move Up ____ 9-17
Move Down ____ 9-17
Move Left ____ 9-17
Move Right ____ 9-18
Move Forward ____ 9-18
Move Back ____ 9-18
Move Distance Slider Scale ____ 9-18
Turn/Orbit Palette ____ 9-19
Orbit Left ____ 9-20
Orbit Right ____ 9-20
Orbit Over ____ 9-20
Orbit Under ____ 9-20
Turn Right ____ 9-20
Turn Left ____ 9-21
Look Up ____ 9-21
Look Down ____ 9-21
Level Off to Camera Position ____ 9-21
Level Off to Camera Target ____ 9-21
Turn/Orbit Angle Slider Scale ____ 9-22
Spaceball ____ 9-23
Setting Up Spaceball ____ 9-23
Spaceball Button Summary ____ 9-23
Using the Spaceball with EPD.Visualizer ____ 9-24
Camera Mode ____ 9-24
Object Edit Mode ____ 9-24

# Database Interface


Creating an ASCII Database ____ A-2
Example of a Database Definition File ____ A-3
Structure of Database Definition File ____ A-7
#DEFINE_DB_FILE Section ____ A-7
#DEFINE_REQUIRED_FIELDS Section ____ A-7
OBJECT_ID FOR DIMENSION III Items (Objects) ____ A-8
#DEFINE_OBJECT Section ____ A-9
DBMS SQL Interface ____ A-11
Verification Tables ____ A-11
Verification Procedures ____ A-14
Automatic Tag Tool ____ A-16
Environment Setup ____ A-16
On all Hosts ____ A-16
On Hosts Running Oracle ____ A-16
On Hosts Running Visualizer ____ A-16
DBMS Demo ____ A-17
Before Using SQL Interface ____ A-17
On All Hosts ____ A-18
On Hosts Running Oracle ____ A-18
On Hosts Running Visualizer ____ A-18
Testing the Interface, Query, and Verification ____ A-18
Testing the Visualizer Interface ____ A-19
Testing the Query Database ____ A-19
Testing Verification ____ A-19


# EPD.Visualizer Textual Graphic File


Textual Graphic File Contents ____ B-2
Textual Graphic File Format ____ B-3
Control Options ____ B-4
DBMS Control Options ____ B-5
Extents Definitions ____ B-6
Color Definitions ____ B-7

Light Definitions _____ B-8
Material Definitions _____ B-10
Path Definitions _____ B-12
GPath Definitions _____ B-13
Object Definitions _____ B-14
Flat Polygon Definition _____ B-16
Shell Definition _____ B-16
Shell Points Definition _____ B-16
Shell Faces Definition _____ B-17
Polyline Definition _____ B-17
Object Definition Example _____ B-18
Gpath Multiple Sectioning _____ B-19

# EPD.Visualizer Binary Graphic File

Binary Graphic File Format _____ C-2
Control Options _____ C-3
Extents Definition _____ C-5
Color Definition _____ C-6
Object Definition _____ C-8
Geometry Specifications _____ C-11
Flat Polygons _____ C-11
Shell Polygons _____ C-12
Polyline _____ C-14
Textual and Binary Graphic File Example _____ C-15
Binary _____ C-16

# Textual and Binary Graphic File Conversion

Syntax _____ D-2
Examples _____ D-2

# DIMENSION III Interface


Introduction _____ E-2

Generating Graphic Files _____ E-3

Processing Single Viewfiles _____ E-3

Processing Multiple Viewfiles _____ E-4

ASCII Graphic File Contents _____ E-6

Error Messages _____ E-7


# CADDS Interface


Introduction _____ F-2

Using the CADDS Interface _____ F-3

Examples _____ F-5

Restrictions _____ F-6

Messages and Errors _____ F-6

Converting Tessellation Data Only _____ F-8

Multiple Input and Output _____ F-9

Assembly Design Trees _____ F-10

Reorienting Objects in Graphic ASCII Files _____ F-11

Options _____ F-11

Examples _____ F-12

cadds2pvsdb _____ F-12


# EPD.Visualizer Movie Player


Getting Started _____ G-3

Loading a Movie _____ G-4

Quitting pvsmovie _____ G-6

Playing a Movie _____ G-6

# Environment Variables


General Behavior ____ H-2
CVDPA_CONFIG_DIR ____ H-2
HOOPS_PICTURE ____ H-2
PVS_ORTHO_MODE ____ H-3
PVS_SECOND_VIEW_SIZE ____ H-3
PVS_STEREO ____ H-3
PVS_STEREO_ANGLE ____ H-3
PVS_TEMP_COLOR ____ H-3
PV_SYSTEM ____ H-4

Graphics Files Loading ____ H-5
CVPATH ____ H-5
PVS_CONCAVE_PERFORM ____ H-5
PVS_CADDS_COLORS ____ H-5

EPD.Visualizer-EPD.Connect Communication ____ H-6
CVDPA_USE_MIRFLAG ____ H-6
OLD_GLOBAL_SPELLING ____ H-6
PVS_EXTERNAL ____ H-6
PVS_CLIENT_FILE ____ H-6
PVS_COLOR_FILE ____ H-7

Path Editor, Path Traversal, and Translation and Rotation ____ H-8
PVS_LOCAL_CPL ____ H-8
PVS_NORMAL_AXIS ____ H-8
PVS_PATHS_DIR ____ H-8
PVS_UNNAMED_REF ____ H-8

PostScript Output ____ H-9
PVS_PRINT_COMMAND ____ H-9
PVS_PRINT_FACTOR ____ H-9

Section Plane ____ H-10
PVS_SECTION_AXIS ____ H-10
CVDPA_SECTIONS_DIR ____ H-10

Zoning _____ H-11
CVDPA_ZONES_DIR _____ H-11
CVDPA_SPACING _____ H-11
Selecting, Scrolling, and Running a Report _____ H-12
PVS_SELECT_SCROLL_REP _____ H-12
CA_ENV_REPORT _____ H-12

# Glossary

# Preface

The *EPD.Visualizer User Guide* describes how to perform the following tasks with the EPD.Visualizer:

- Converting graphics files from a variety of formats into Visualizer textual graphic file format.
- Assembling CAD models from Visualizer textual graphic files.
- Adjusting lighting and surface effects.
- Visually entering CAD models and making movies that you can play back later.
- Performing static sectioning and interference checking.
- Defining paths that include orientation specifications through CAD models.
- Moving objects and sectioning planes on paths through CAD models to perform dynamic interference checking and sectioning.

## Related Documents

The following documents may be helpful as you use *EPD.Visualizer User Guide*:

- *Installing EPD.Visualizer*
- *EPD.Connect User Guide*

## Book Conventions

The following table illustrates and explains conventions used in writing about Optegra applications.

| Convention | Example | Explanation |
|---|---|---|
| EPD_HOME | `cd` $EPD_HOME`/install` (UNIX)<br><br>`cd` %EPD_HOME%`\install` (Windows) | Represents the default path where the current version of the product is installed. |
| Menu selections | Vault > Check Out > Lock | Indicates a command that you can choose from a menu. |
| Command buttons and options | Mandatory check box, Add button, Description text box | Names selectable items from dialog boxes: options, buttons, toggles, text boxes, and switches. |
| User input and code | `Wheel_Assy_details`<br><br>`-xvf /dev/rst0`<br><br>`Enter command>` **`plot_config`** | Enter the text in a text box or on a command line.<br><br>Where system output and user input are mixed, user input is in bold. |
| System output | `CT_struct.aename` | Indicates system responses. |
| Parameter and variable names | `tar -cvf /dev/rst0` filename | Supply an appropriate substitute for each parameter or variable; for example, replace filename with an actual file name. |
| Commands and keywords | The `ciaddobj` command creates an instance of a binder. | Shows command syntax. |
| Text string | `"SRFGROUPA"` or `'SRFGROUPA'` | Shows text strings. Enclose text strings with single or double quotation marks. |
| Integer | n | Supply an integer for *n*. |
| Real number | x | Supply a real number for *x*. |
| # | `# mkdir /cdrom` | Indicates the root (superuser) prompt on command lines. |
| % | `%` `rlogin` remote_system_name `-l root` | Indicates the C shell prompt on command lines. |
| $ | `$` `rlogin` remote_system_name `-l root` | Indicates the Bourne shell prompt on command lines. |
| > | `> copy` filename | Indicates the MS-DOS prompt on command lines. |
| Keystrokes | `Return` or `Control-g` | Indicates the keys to press on a keyboard. |

## Online User Documentation

Online documentation for each Optegra book is provided in HTML if the documentation CD-ROM is installed. You can view the online documentation from an HTML browser or from the HELP command.

You can also view the online documentation directly from the CD-ROM without installing it.

### From an HTML Browser:

1. Navigate to the directory where the documents are installed. For example,

   $EPD_HOME`/data/html/htmldoc/` (UNIX)

   %EPD_HOME%`\data\html\htmldoc\` (Windows NT)
2. Click `mainmenu.html`. A list of available Optegra documentation appears.
3. Click the book title you want to view.

### From the HELP Command:

To view the online documentation for your specific application, click HELP. (Consult the documentation specific to your application for more information.)

### From the Documentation CD-ROM:

1. Mount the documentation CD-ROM.
2. Point your browser to:

   CDROM_mount_point`/htmldoc/mainmenu.html` (UNIX)

   CDROM_Drive:`\htmldoc\mainmenu.html` (Windows NT)

## Printing Documentation

A PDF (Portable Document Format) file is included on the CD-ROM for each online book. See the first page of each online book for the document number referenced in the PDF file name. Check with your system administrator if you need more information.

You must have Acrobat Reader installed to view and print PDF files.

The default documentation directories are:

- $EPD_HOME`/data/html/pdf/`doc_number`.pdf` (UNIX)
- %EPD_HOME%`\data\html\pdf\`doc_number`.pdf` (Windows NT)

## Resources and Services

For resources and services to help you with PTC (Parametric Technology Corporation) software products, see the *PTC Customer Service Guide*. It includes instructions for using the World Wide Web or fax transmissions for customer support.

## Documentation Comments

PTC welcomes your suggestions and comments. You can send feedback in the following ways:

- Send comments electronically to `doc-webhelp@ptc.com`.
- Fill out and mail the PTC Documentation Survey located in the *PTC Customer Service Guide.*

Chapter 1

# Introduction to EPD.Visualizer

This chapter introduces you to EPD.Visualizer and its basic concepts. It also explains Visualizer features and usage. The topics discussed are:

- What Is EPD.Visualizer?
- Basic Concepts
- Using the EPD.Visualizer Window
- Color Editor
- Permanent Mouse Dynamics
- Starting an EPD.Visualizer Session

# What Is EPD.Visualizer?

EPD.Visualizer is advanced visualization software that lets you create CAD models from graphical information generated by a variety of CAD systems. You can enter the CAD models visually and traverse paths through them. You can record animated movies as you move through the models and play them back later. EPD.Visualizer can simulate virtual walk-throughs of 3D scenes such as power or process plants, offshore platforms and ships, manufacturing and office facilities, and civil sites. These 3D scenes can be accented with lights and specific material surfaces (such as chrome) to create shadows and reflection for greater realism. EPD.Visualizer can also access the Database Management System (DBMS) for live engineering data through the SQL interface. EPD.Visualizer lets you analyze of complex models through sectioning, interference checking, and path traversal. You can export some EPD.Visualizer sections and other results to CAD systems.

You can examine expansive and equally complex 3D scenes of jets, automobiles, or engine assemblies for the inspection of the complete assembly, subassemblies, and individual components. In addition, you can simulate high-quality presentations with clarity, detail, and graphics performance with the compatible Hewlett-Packard (HP), Sun, Silicon Graphics, Digital, and IBM workstations to assess aesthetics, clearances, construction or manufacturing requirements, and overall site conditions.

In a matter of minutes you can easily transfer information from many CAD lines to EPD.Visualizer for on-screen viewing. Models from DIMENSION III, CADDS, MEDUSA can be accessed transparently through the EPD.Visualizer interface, which uses a common graphic file format.

Please note: The MEDUSA Process-Equipment Design System (MPDS) interface is documented in the INFO file of the most current MPDS release, beginning with Release 12.2.

## How Can You Use EPD.Visualizer?

Designers, engineers, facility owners, contractors, and members of regulatory agencies can inspect complete CAD models at any stage of development. You can view conceptual alternatives, construction-ready designs, or collections of existing CAD models quickly and easily. In an instant, you can compute and display views of even the most complex project or mechanical assembly from any distance and at any angle.

You can become more efficient by using this state-of-the-art software for everyday tasks or to communicate your work product to clients by conducting:

- Walk-throughs of proposed projects
- Preliminary site planning or general arrangement studies
- Internal design reviews between project teams and discipline groups
- Simulation of equipment installation and removal
- Human engineering studies
- Operator training
- Shutdown and disassembly simulations and studies
- Disaster planning and simulation
- Sales presentations

No special skills or training are required to use EPD.Visualizer. Engineering, management, purchasing, and scheduling personnel, as well as subcontractors, can access entire 3D scenes or portions of interest with minimal instruction.

## EPD.Visualizer Features

After converting your CAD models to a EPD.Visualizer textual or binary graphic file and adding the graphics in EPD.Visualizer, you can perform the following functions:

- Defining paths that include orientation specifications through CAD models.
- Traversing either the camera or an object (such as a person, crane, or forklift) on paths to simulate a construction or maintenance sequence, including an animated movie sequence.
- Traversing an object on paths to check collisions.
- Sectioning a model with a static plane.
- Traversing a sectioning plane with or without an attached object on a path and displaying the resulting sections simultaneously.

- Measuring distances in sections with dimensioning tools.
- Defining a sequence of views including annotations for animated playback or later presentation or editing.
- Displaying database information about an object in the 3D scene for ongoing maintenance to reduce off-site inspection and data gathering trips.
- Resolving potential conflicts using interference checking before construction or assembly begins.
- Evaluating spatial requirements by moving and orbiting in any direction to view the 3D scene from every angle and zooming in for a closer look.
- Controlling the location of the camera and its ultimate target.
- Using context-sensitive online help to obtain the assistance for each button, tool, menu, or dialog box, as well as quick reference help on the palettes.
- Translating, rotating, hiding, or deleting items for a better view of your 3D scene or other surrounding items.
- Creating lighting to add dimension and realism to the 3D scene.
- Selecting rendering features to visualize different finishes, such as matte or mirrored, radiosity, ray tracing, and shadowing for realistic evaluations.

## Advanced Rendering Module

This manual documents the advanced rendering features - radiosity and ray tracing. If you have not purchased the advanced module, the items on the menus are grayed out; however, the basic rendering features are available.

# Basic Concepts

This section provides the basic concepts you need to understand EPD.Visualizer:

- System Overview
- Neutral Graphic Files
- Metafiles
- 3D Viewing

While using EPD.Visualizer, refer to this section for an explanation of how EPD.Visualizer works. The glossary provides descriptions of terms used in EPD.Visualizer.

## System Overview

EPD.Visualizer is designed so you can merge information from several CAD systems for simultaneous display of a complete 3D scene. You can process DIMENSION III, CADDS, MEDUSA, and SYSTEM 9 models or parts using the interfaces documented in the following appendixes:

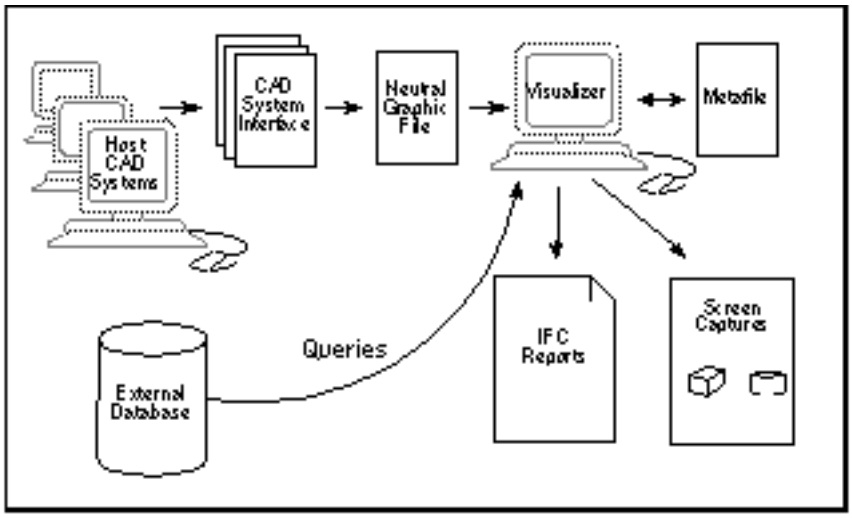
- Appendix E, “DIMENSION III Interface”
- Appendix F, “CADDS Interface”

These interfaces generate a neutral graphic file format, either textual or binary, and are documented in Appendix B, “EPD.Visualizer Textual Graphic File”and Appendix C, “EPD.Visualizer Binary Graphic File”. With these graphic files, you can effectively merge models from several CAD systems enabling you to explore and visualize complete 3D scenes.

In addition to the exploration aspects of EPD.Visualizer, you can also perform interference studies and produce reports, access external database information, and capture views for printing and viewing outside of EPD.Visualizer.

Figure 1-1 shows an overview of EPD.Visualizer and how you can process and manage information:

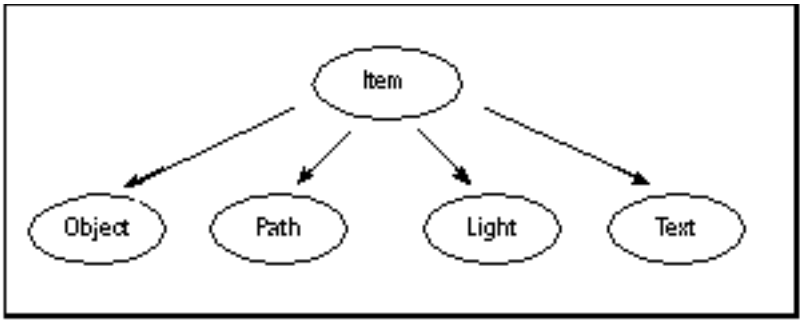
**Figure 1-1 EPD.Visualizer Overview**




## Objects and Items

In EPD.Visualizer, the term object refers to the design geometry that you import from your CAD system. The term item refers to something that is part of your 3D scene within EPD.Visualizer. As Figure 1-2 shows, items can include objects, paths, lights, and text or annotation.

**Figure 1-2 Relationship Among Items in EPD.Visualizer**



# Neutral Graphic Files

You can add any number of textual and binary neutral graphic files during an EPD.Visualizer session. A textual graphic file contains the definitions of geometric objects, colors, lights, materials, and paths in EPD.Visualizer ASCII format. You can edit or view this file using a standard text editor. The host CAD system typically appends a `.gaf` (graphic ASCII file) extension to the textual file name.

A binary graphic file is an encoded file that defines geometric objects and colors in EPD.Visualizer binary format. These files are extremely compact and can be processed many times faster than textual graphic files. However, unlike a textual file, you cannot edit or view this file as readable text. The host CAD system typically appends a `.gbf` (graphic binary file) extension to the binary file name.

You can convert textual graphic files to binary graphic files, and binary graphic files to textual graphic files. For further information, refer to Appendix B, “EPD.Visualizer Textual Graphic File”and Appendix C, “EPD.Visualizer Binary Graphic File”.

After you add a graphic file to EPD.Visualizer, each subsequent graphic file added is automatically converted to the units of the first graphic file (such as feet, miles, or meters). For example, if you add a graphic file defined in meters, then add a graphic file defined in inches, and another defined in millimeters, the last two graphic files added are converted to meters.

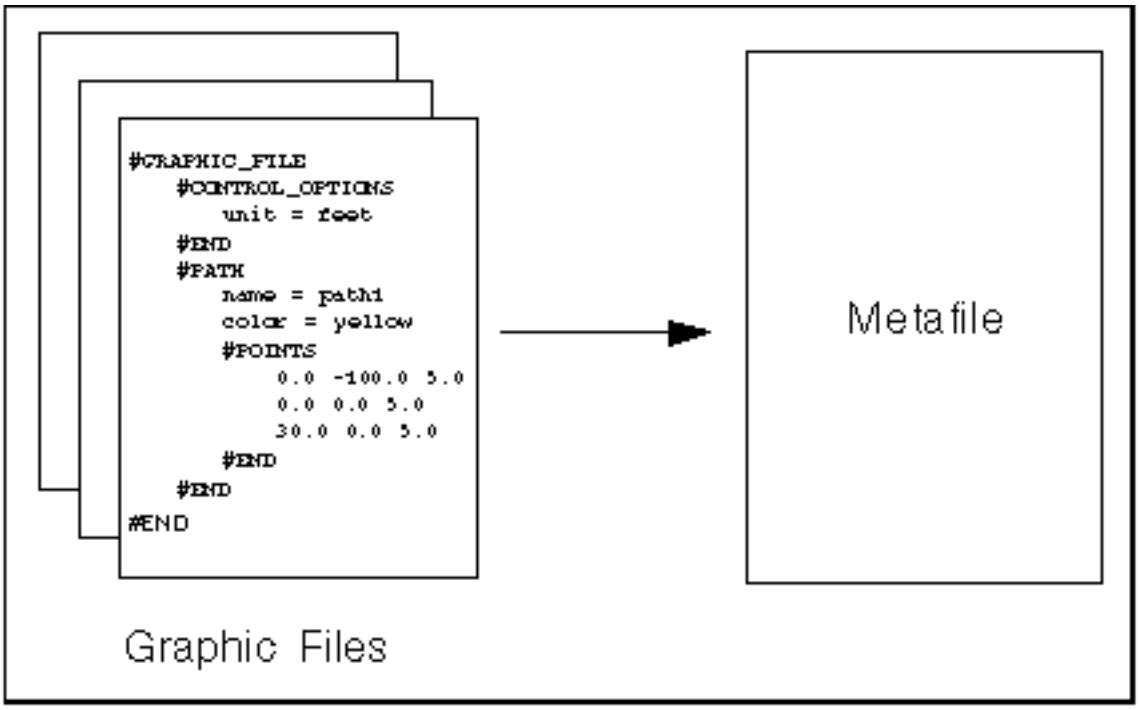
You can also work in different units by setting the working units. When you perform walk-through operations, item traversal, or item verification, all calculations are displayed in the working units you select.

## Metafiles

A metafile is an ASCII file that contains a complete description of an EPD.Visualizer session. This includes all geometric objects, views, view sequences, lights, paths, color definitions, materials, notes, and control settings, such as working units and slider scale settings.

If you add several textual and binary graphic files to an EPD.Visualizer session, you can save the graphic files into one metafile, as shown in Figure 1-3:

**Figure 1-3 Graphic File and Metafile Relationship**




When you save a metafile, if you do not add an extension, EPD.Visualizer appends a `.mf` extension to the metafile name.

## 3D Viewing

After you become familiar with the coordinate system and general viewing concepts used in EPD.Visualizer, you can easily manage your 3D scene. These spatial concepts are described in this section.

### Coordinate System

The global environment used in EPD.Visualizer is based on the Cartesian coordinate system with a z-up vertical orientation. Any walk-through operations you perform are calculated with respect to this xyz system.

If your graphic files are based on a y-up coordinate system, you can select the option, Y Axis is Vertical, when you add your graphic file. After you select this option, EPD.Visualizer converts the graphic file to the z-up coordinate system.

## Viewing

Your eye is the camera position and what you see in the 3D scene is within view of the camera lens. The camera target point is always located at the center of the graphics window at a depth you specify. All walk-through operations are based on this concept of the camera.

## Linear and Angular Movement

EPD.Visualizer enables you to instantly change the camera position and camera target to view any aspect of your 3D scene. You can quickly shift the camera in a linear or an angular direction.

You can move in a linear direction by a specific number of units, using the buttons on the Move palette. You can move up or down, left or right, and forward or back. For example, if you click the Move Forward button, the camera moves forward so you are drawn closer to the 3D scene. Likewise, if you click the Move Left button, the camera position moves to the left.

You can move in an angular direction about the vertical axis of the 3D scene by a specific number of degrees, using the Turn and Orbit buttons. You can turn left or right, look up or down, and orbit left, right, over or under. For example, if you click the Orbit Left button, the camera position rotates left.

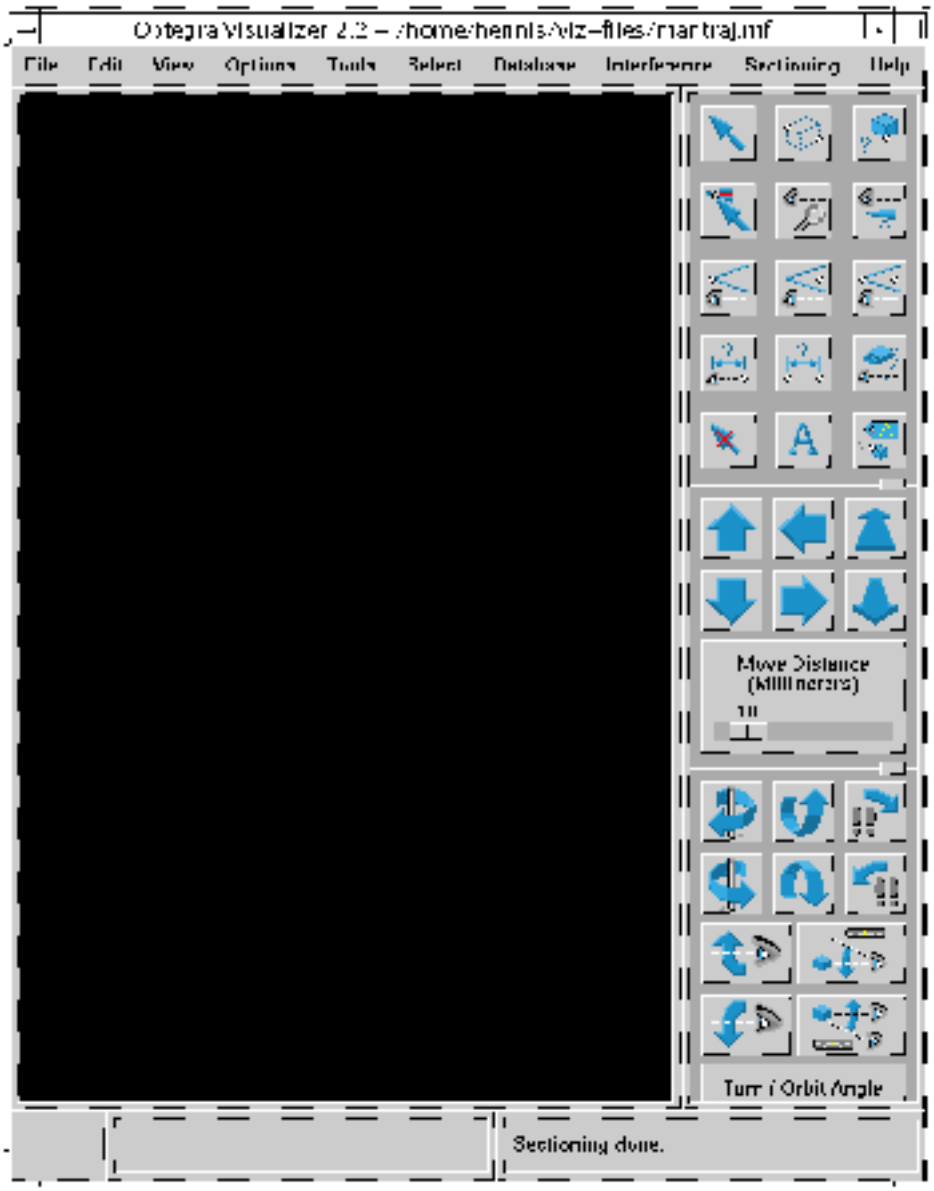
You can control the rendering of the 3D scene while you are using the walk-through buttons. The following describes these methods:

- To move the camera rapidly through the 3D scene without waiting for a complete rendering of all graphics, click any of the walk-through buttons before the 3D scene is completely displayed.
- To suppress the rendering of the 3D scene while using any of the walk-through buttons, hold down the Shift key while clicking the button.
- To change the Move Distance or Turn/Orbit Angle during rendering of the 3D scene, click the slider scales.

# Using the EPD.Visualizer Window

This section describes the EPD.Visualizer window, explains how to navigate within EPD.Visualizer with the mouse, and generally describes how to use the interface.

In addition to standard windowing features, the EPD.Visualizer window has specific features as shown below.




The following sections explain the specific features of the EPD.Visualizer window.

## Graphics Window

Most of the EPD.Visualizer window is occupied by the graphics window. Your 3D scene is displayed in the graphics window. To the right of the graphics window are three palettes. Beneath the graphics window are the prompt window (on the left) and the status window (on the right).

## Prompt and Status Windows

The prompt window displays prompts that require a response from you.

The status window displays the status of operations as you perform them.

## Palettes

Three palettes are located vertically alongside the graphics window:

- Tool Palette
- Move Palette
- Turn/Orbit Palette

Slider scales are found in palettes and dialog boxes and are explained in this section.

The prompt window displays a brief description of each tool and button when you place the mouse cursor to a tool, button, or the slider scale.

Customize the layout of your EPD.Visualizer window with the following menu items:

- Move the palettes to the left or right side of the graphics window by selecting Options > Left/Right Palette
- Hide or show the palettes by selecting Options > Hide/Show Palette
- Vertically stretch or shrink the palettes with the slider located on the bottom right of the palette.

You can control the rendering of the 3D scene while you are using the Move or Turn/Orbit Palette buttons. For further information see “Linear and Angular Movement” on page 1-9. The following describes these methods:

- To move the camera rapidly through the 3D scene without waiting for a complete rendering of all graphics, click any of the walk-through buttons before the 3D scene is completely displayed.
- To suppress the rendering of the 3D scene while using any of the walk-through buttons, hold down the Shift key while clicking the button.
- To change the Move Distance or Turn/Orbit Angle during rendering of the 3D scene, click the slider scales.

## Tool Palette and Other Palettes

The tools in the Tool Palette require action with the mouse in the graphics window. A brief description of each tool is displayed in the prompt window when you place the pointer on it.

To pick up a tool, use the following steps

1. Click on the tool to highlight it.
2. Move the pointer into the graphics window.

   The name and prompt for the tool are displayed in the prompt window.
3. Follow the directions displayed in the prompt window to use the tool.
4. Click the tool to clear it when you finish using it.

Please note: Tools remain active until you clear them or until you select other tools.

Move Palette: The buttons in the Move Palette enable you to move in a linear direction. Control the distance you move with the Move Distance slider scale.

Turn/Orbit Palette: The buttons in the Turn/Orbit Palette enable you to move in an angular direction. Control the angle you move with the Turn/Orbit Angle slider scale.

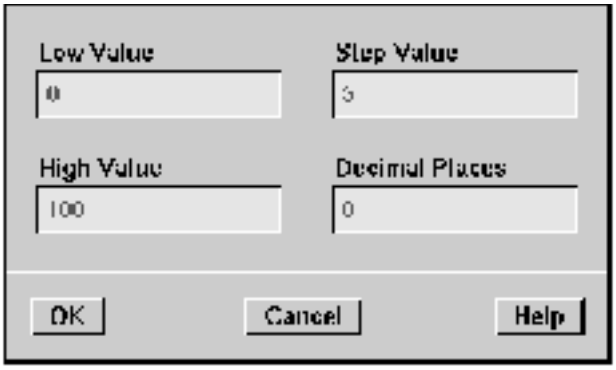
## Slider Scales

Slider scales are found in palettes and dialog boxes. Atypical slider scale is shown below:

Use one of the following for the slider scale:

- Click the bar on the slider scale to increase or decrease by the step value (a changeable value), the following text explains on how to customize the slider scales.
- Drag the slider to increase or decrease.

To customize the slider scales, click the right mouse button on the slider scale value, and the Scale Customization dialog box is displayed. An example of the Scale Customization dialog box is shown below:



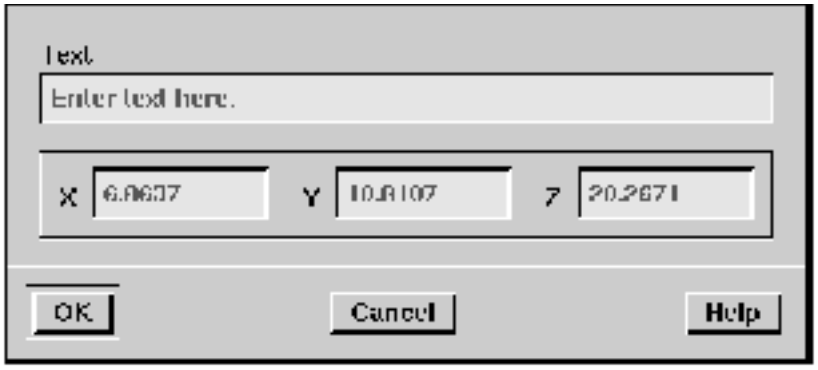

Enter a new value for the low or high value and click OK.

Step Value: The number of units the slider scales increase or decrease when you click the bar. For example, if you enter a step value of 15 and click the bar, the scale value increases or decreases by 15. Enter a new value and click OK.

Decimal Places: You can change the value displayed in the slider scale to decimal form. Enter a number between one and three to specify the number of digits that will be displayed to the right of the decimal point and click OK.

## Text Editor

A Text Editor dialog box is displayed when you use the Text or Object Tagging tools. The Text Editor also displays the xyz coordinates of the digitize or object you selected. Before entering text, select a different location without closing the Text Editor. Click the in the text box to enter text. An example of the Text Editor dialog box is shown below:




## Menu Bar

The menu bar is the horizontal bar at the top of the screen and contains menus represented by names.

Menus: To select a menu, use one of the following procedures:

- Click a menu name and drag your pointer down the menu. Then release the mouse button on the appropriate menu item.
- Click the menu name to open the menu, which remains open until you select a menu item.

To close a menu, slide off the menu to close it or click outside the menu area.

Menu Items: The menu items are displayed when you select a menu name in the menu bar.

- Menu items become available depending on the operations you perform. If a menu item is grayed out, it is not available.
- If you select a menu item that is followed by an ellipses (...), a dialog box is displayed. If buttons on dialog boxes are grayed out, they are not available.
- If you select a menu item that is followed by a filled right arrow, an additional menu is displayed.
- If you click the diamond-shaped buttons next to a menu item, you can switch between the available options. If the button is filled, it is selected.

- If you click the square-shaped buttons next to a menu item, you can select one or more options. If the button is filled, it is selected.
- If a menu item has no symbol next to it, the particular action takes effect as soon as you select it. For example, if you select Edit > Hide, selected items in your 3D scene instantly disappear from view.

## Dialog Boxes

When you select menu items followed by ellipses, a dialog box is displayed. The dialog box provides options that enable you to perform numerous tasks, such as setting attributes, changing colors, and defining properties. Each user can define the screen positions of dialog boxes when they open; for an explanation of this feature, see "CVDPA_CONFIG_DIR" on page H-2 .

Some dialog boxes might require that you make a selection or interact with it before proceeding, which is indicated by a beep.

The following sections describe items contained in the dialog box or that are displayed as a result of an action within the dialog box.

### Text Box

Click inside the text box to enter text. EPD.Visualizer is not case-sensitive. You can use any alphabetic characters, numbers 0-9, underscores, and spaces in the text boxes.

### Lists

Entries such as file names, colors, or fonts are displayed in lists. You can select multiple entries or one entry at a time, depending on the dialog box.

To scroll through a list

- Click above or below the vertical scroll bar to move up or down the list. Click to the right or left of the horizontal scroll bar to shift the list from side to side.
- Use the pointer to drag the vertical scroll bar up or down to move the list vertically, or drag the horizontal scroll bar from side to side to shift the list horizontally.
- Click inside the list and press the UP or DOWN arrow to move up or down the list, line by line.

To select multiple entries in a list either

- Click and drag the pointer from one entry in the list down the list to select consecutive entries.
- Hold down the `Control` key and click random entries.

## Option Buttons

Click the diamond-shaped buttons to switch between the available options. Click the square-shaped buttons to select one or more options. If the button is filled, it is selected.

## Tools

Tools sometimes appear in a dialog box and require that you perform an action with the mouse in the graphics window. Use a tool in a dialog box the same way as the tools in the Tool Palette. For further information, see “Palettes” on page 1-11.

## Buttons

The following are some of the buttons found in dialog boxes and their functions:

- Help — Displays on-line help for the dialog box
- OK — Accepts the current change(s) and closes the dialog box
- Accept — Accepts the current change(s)
- Apply — Applies the current change(s) in your 3D scene and the dialog box remains open
- Add — Adds a selected graphic file
- Open — Opens the selected metafile
- Filter — Updates a file selection dialog box with directories and files; refer to the following section for an explanation on how to use the Filter
- Save — Saves a graphical element such as a section or Gpath in an EPD.Visualizer textual graphics file.
- Close — Closes the dialog box

## File Selection Dialog Box

This section explains the dialog box that is displayed when you are selecting or saving files and describes the following options:

- Filter text box
- Directories list
- Files list
- Selection text box

An example of the File Selection dialog box is shown below:

Filter: Displays the current directory.

Directories: Displays existing directory names.

Files: Displays existing file names.

Selection: Displays the path name if you use the filter. Otherwise, this text box is empty.

Use the following steps for the Filter button:

1. Enter a path name with or without wild card symbols in the Filter text box to display a specific list of directory or file names.
2. Click Filter. The existing directories or files in the path name you entered are displayed.

   For example, to display other directories and all of the files contained in the directory Samples list, filter the following path name:

```
/HP/Drawing/Samples
```

   To display all the files with a `.gaf` extension in the Samples directory, filter the following path name with the wild card symbol and extension:

```
/HP/Drawing/Samples/*.gaf
```

To select from the Directories list, double click the appropriate directory name, and the names of any files in that directory are displayed in the Files list.

To select from the Files list, double click the appropriate file name, and the name is displayed in the Selection text box.

Use the following steps for the Selection text box:

1. Enter the directory and/or file name.
2. Click OK, Add, Save, or Open.

## Slider Scale

Refer to “Slider Scales” on page 1-13 for more information.

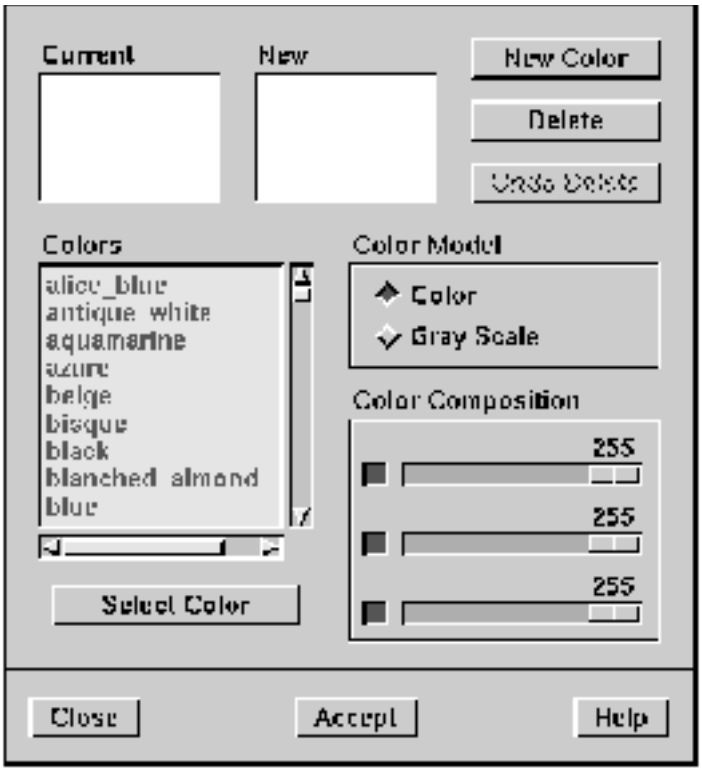
# Color Editor

The Color Editor enables you to specify new colors for planar surfaces, the graphics window, sections, Gpaths, text, and lights. As you define or select colors, the current color and a new color is displayed in the dialog box.

You can define a new color with the Color Composition slider scales after selecting a Color Model option. If you select the Color option, you can define a new color using the RGB (red, green, and blue) scales. If you select the Gray Scale option, you can use the RGB scales to define a new color in shades of black and white.

You can also select a color from the list of Colors. In addition, you can use the Select Color tool to verify or select a color from geometry in the 3D scene.

The Color Editor dialog box is shown below:




## Defining a New Color

Use the following steps to define a new color:

1. Select the Color Model (Color or Gray Scale).
2. Drag the sliders on the Color Composition slider scales to adjust the color.

3. Click New Color and a text box is displayed.
4. Enter a name (up to 30 characters) in the text box.
5. Click Accept to save the color.

## Deleting a Color

Use the following steps to delete a color:

1. Click the color in the Colors list.
2. Click Delete, and the color is removed from the Colors list and temporarily stored. You can restore a deleted color with Undo > Delete.

## Restoring a Deleted Color

To restore a deleted color, click Undo > Delete. You cannot restore the deleted color after you close the dialog box, define a new color, or edit a color name.

## Selecting or Verifying a Color

Use the following steps to select or verify a color from geometry in your 3D scene:

1. Pick up the Select Color tool and select an item in the 3D scene with the desired color.
2. Click Accept if you want to use the selected color.

# Permanent Mouse Dynamics

You can perform view manipulations by moving the mouse pointer on the viewing area. You can use all three mouse buttons to perform such view manipulations as screen axis motions, model axis motions, and quick view centering. These operations are always available, except when the Mouse Viewing tool in the Tool Palette is active and with Permanent Zoom and Temporary Zoom.

## Screen Axis Motions

Screen axis motions allow you to translate the camera on the screen axis, to zoom and unzoom, and to rotate around any screen axis, including the one perpendicular to the screen. You can enable screen axis motions at any time, whether or not a tool is active, by dragging the mouse with the middle or right mouse button.

The screen axis motions, and the actions you need to perform to activate them, are described in the following table:

**Table 1-1 Screen Axis Motions**

| Desired Motion | Actions You Perform |
|---|---|
| Scroll | Drag with the middle button to translate the camera on the horizontal and vertical screen axis so that the scene adheres to the mouse pointer. |
| Zoom | Drag vertically with the middle mouse button while pressing the `Shift` key:<br>Dragging up zooms the camera on the scene.<br>Dragging down unzooms the camera away from the scene. |
| Rotation | Drag with the right mouse button:<br>Dragging vertically orbits the camera around the vertical screen axis so that the camera rotates around the scene.<br>Dragging horizontally orbits the camera around the Z model axis so that the scene rotates around Z to the pointer. (It is preferable to use the Z model axis rather than the vertical screen axis.) |
| Third Rotation | Drag with the right mouse button while pressing the `Shift` key:<br>Dragging left or up rolls the camera around its axis so that the scene rotates counter-clockwise in the screen plane.<br>Dragging right or down rolls the camera around its axis so that the scene rotates clockwise in the screen plane.<br>Note: These actions are meaningful when the pointer is on the top or right. |

## Behavior of Screen Axis Motions

The scene updates each time you drag the mouse to perform a new motion. If any previous updates are still occurring, these are interrupted and the scene flashes. A mouse drag is effective only when you move the mouse pointer at least the number of pixels indicated by the Threshold scale in the User Preferences dialog box. The User Preferences dialog box is available from the Options menu (Options > Preferences).

EPD.Visualizer obtains the steps for each of the screen axis motions in the following way:

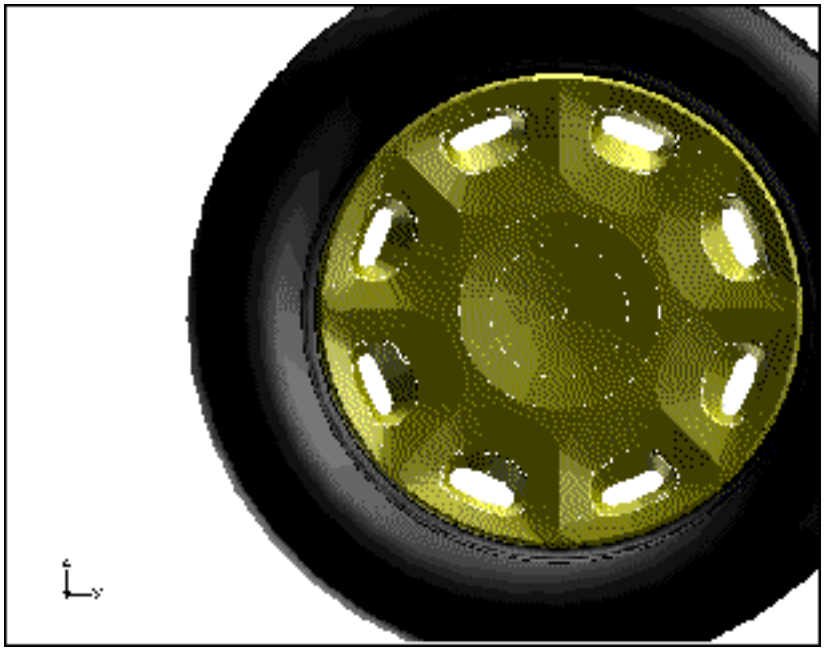
- The steps used for screen axis translations come from the view field. With translations, the camera translates on a horizontal and vertical screen axis so that the scene adheres to the mouse pointer.
- The steps used for zoom and unzoom are computed proportionally from the view field. (The view field is the width and height of the visible parts of the scene as viewed through the camera.)
- The steps used for rotations come from the Turn/Orbit Angle slider scale in the Turn/Orbit Palette. The angle is initially set to 15 degrees. All rotation steps are an integer times this angle. During rotations, the pixel threshold is twice the normal one to avoid unwanted rotations. For more information about the Turn/Orbit Angle slider scale, see Chapter 9, “Palettes.”

The total motion that has occurred since your last mouse button action is displayed in the status window.

## Example of Screen Axis Motions

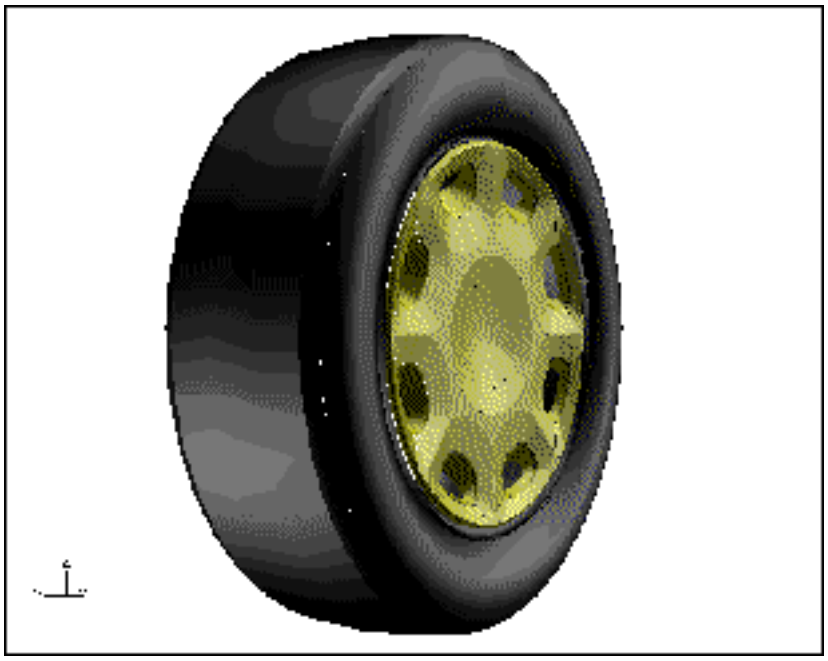
Suppose you want to digitize two points that cannot be visible together in the same view. To do so, perform the following steps:

1. Load the graphics files. After you load the files, the main view looks as follows:

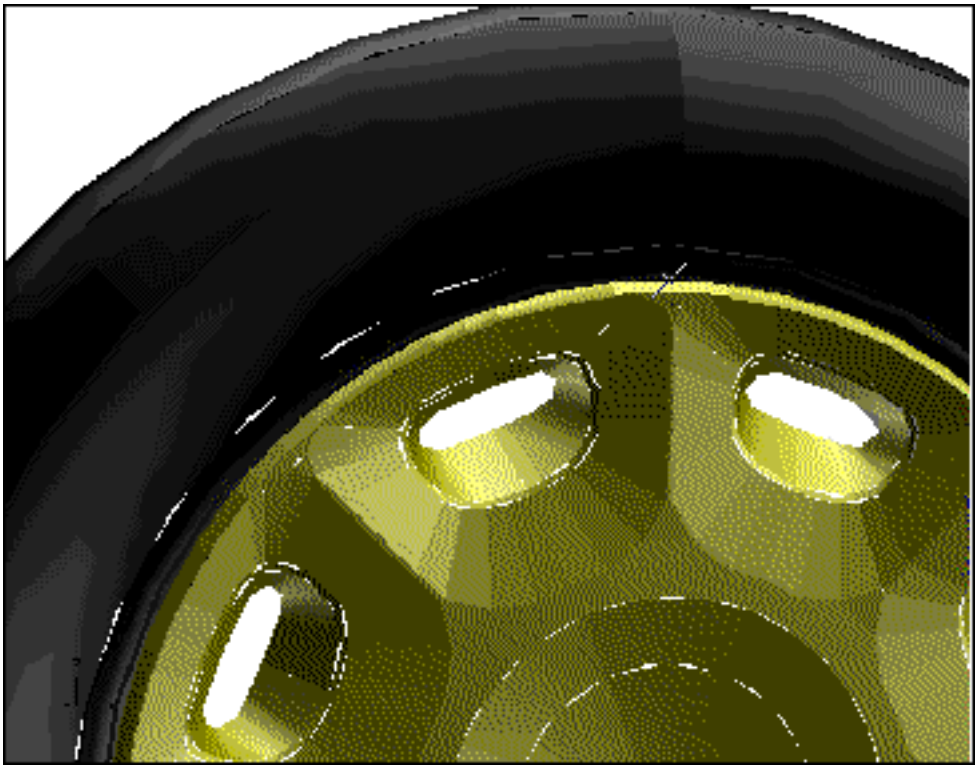


2. Drag down with the middle mouse button while pressing the `Shift` key. (Do this without selecting or deactivating any tool.) The view quickly unzooms so you can view the whole scene.
3. Drag left with the middle mouse button to quickly center the scene.

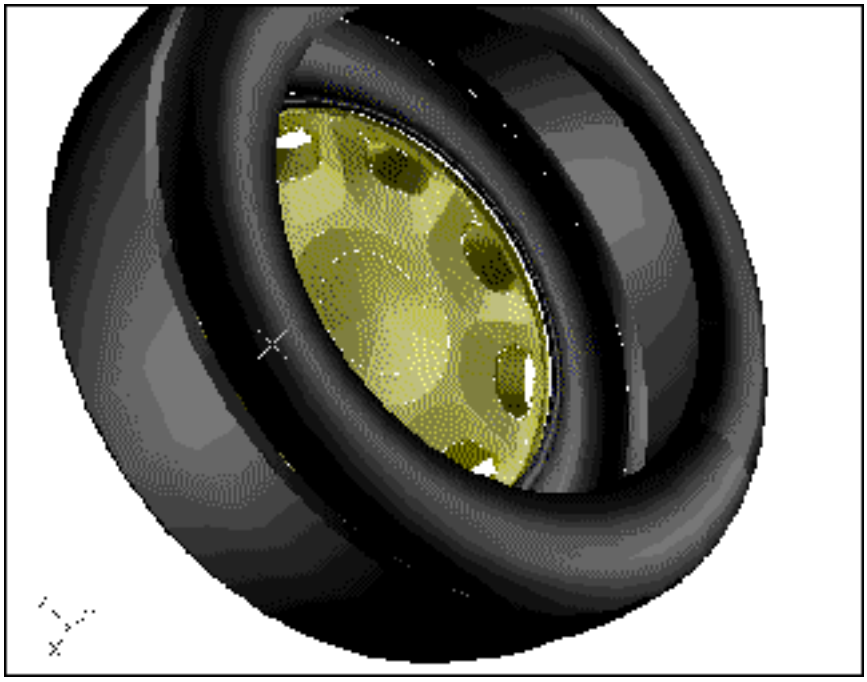
4. Drag left with the right mouse button to rotate the scene and obtain the following view:



5. Perform additional rotation and zoom operations to obtain an ideal view for digitizing the first point. Now you can accurately digitize a point between the rim and the tire, as show below:



6. Perform additional rotation, translation, and unzoom operations to focus the view on the reverse side of the tire. Now you can digitize a second point that was not visible in the previous view, as shown below:



# Model Axis Motions

Model Axis motions allow you to translate the camera on a model axis or rotate it around a model axis. You can enable the model axis motions at any time, whether or not a tool is active, by clicking the middle or left mouse button. A second click on either of these buttons disables model axis mode.

To perform a model axis motion, you need to select an axis. When you click the middle or right mouse button, the axis in the bottom left corner appears in yellow and the X axis appears in red. The axis that is displayed in red is the selected axis that will be used for the motion. The X axis is the default. You can select another axis by clicking with the left mouse button on the line or letter for that axis in the bottom left corner. If you click anywhere else with the left mouse button, the current axis cycle is selected.

The model axis motions, which move or orbit the camera along the selected axis, and the actions you need to perform to activate them, are described in the following table:

**Table 1-2 Model Axis Motions**

| Desired Motion | Actions You Perform |
|---|---|
| Move the camera | Drag with the middle mouse button:<br>Dragging right or up translates the scene positively on the axis.<br>Dragging left or down translates the scene negatively on the axis. |
| Orbit the camera | Drag with the right mouse button:<br>Dragging right or up rotates the scene positively around the axis.<br>Dragging left or down rotates the scene negatively around the axis. |

When you release one of the mouse buttons, and are not pressing the `Shift` key, the model axis motions are disabled and the model axes recover their initial colors. If you release a mouse button, and at the same time press the `Shift` key, the model axis motions remain enabled. You can select another axis or perform another motion.

## Behavior of Model Axis Motions

The behavior of model axis motions is similar to screen axis motions. EPD.Visualizer obtains the steps used for translations from the view field. It obtains the steps used for rotations from the Turn/Orbit Angle slider scale in the Turn/Orbit Palette. The angle is initially set to 15 degrees. All rotation steps are an integer times this angle. During rotations, the pixel threshold is twice the normal one to avoid unwanted rotations. For more information about the Turn/Orbit Angle slider scale, refer to Chapter 9, “Palettes.”

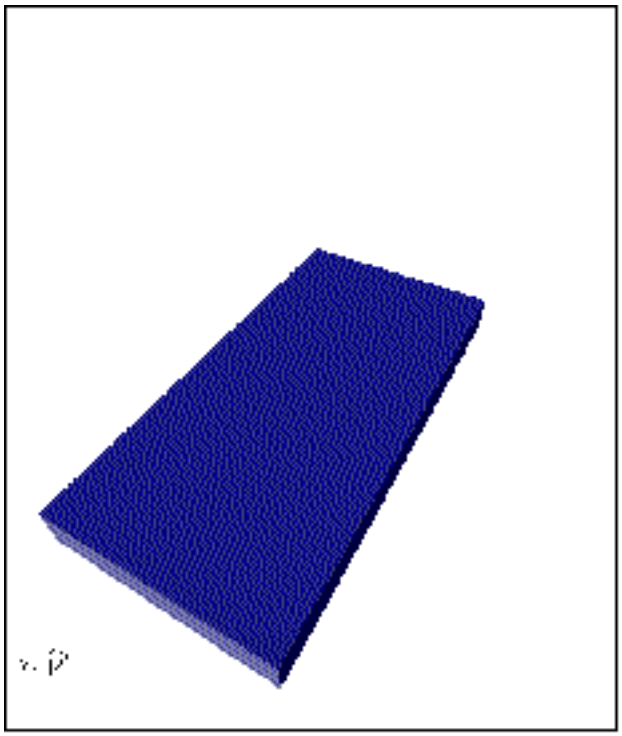
The total motion that has occurred since your last mouse button action is displayed in the status window.

## Example of Model Axis Motions

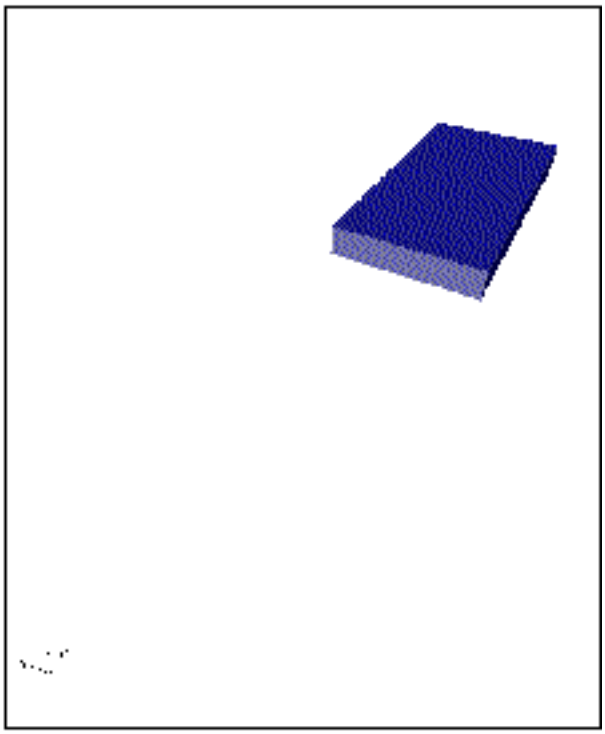
Suppose you want to move the camera so that the scene appears to translate on an axis that is oblique and does not correspond to a screen axis. To do so, perform the following steps:

1. Click the middle or right mouse button. (Do this without selecting or deactivating any tool.) The X axis appears in red. The Y and Z axes appear in yellow.
2. Drag up or right with the middle mouse button to translate the scene positively along the X axis. When you release the mouse button, the axes recover their initial colors and EPD.Visualizer resumes its current operation.

The scene as it appears before and after you move the camera is show below:



Before



After

## Quick View Centering

Quick view centering allows you to quickly center the view on a point in the viewing area. It allows you to center a view more quickly and accurately than by using screen axis translations. This centering is only available when no tool is active, such as at startup or when a tool has been deactivated.

You center a view by clicking the left mouse button on a point in the viewing area, even on a point where there is no geometry. By digitizing the point, you to set the current camera target, called the orbit point, to center all camera rotations. The digitized point flashes rapidly and the scene is centered on it. You set the digitized point as the new orbit point in a way similar to the Set Camera Target tool available in the Tool Palette. For information about the Set Camera Target tool, refer to Chapter 9, “Palettes.”

## Considerations

Some restrictions and limitations that apply to permanent mouse dynamics are the following:

- To explicitly set the orbit point, you must first exit an active tool.
- Whenever axis display is disabled, it is automatically enabled when you begin model axis motions to allow axis selection.
- The directions described in this section for using model axis motions may, for some views, seem unnatural to you. This is because it is impossible to orient properly spaced axes in a plane.

# Starting an EPD.Visualizer Session

This section helps you to start a EPD.Visualizer session. You should read the following sections before you begin:

- Adding graphic files or opening metafiles
- Starting a new session
- Quitting EPD.Visualizer

## Running EPD.Visualizer

After you install EPD.Visualizer and the its license manager, refer to the on-line release notes available from the Help menu pulldown for important information that might pertain to your system.

Please note: You cannot start EPD.Visualizer without proper licensing. Before proceeding, be sure you create a license file as described in the *Installing EPD.Connect, EPD Roles, and EPD.Visualizer.*

## Adding Graphic Files or Opening Metafiles

Before you can begin using EPD.Visualizer, you must first add a graphic file, either textual or binary, or open a metafile under the File menu. You can add sample graphic file(s) or open a sample metafile located in the `common/data` directory.

Please note: Textual graphic files are appended with a `.gaf` extension, binary graphic files are appended with a `.gbf` extension, and metafiles are appended with a `.mf` extension.

After you add the graphic file(s) or open a metafile and the 3D scene is displayed in the graphics window, you can start using EPD.Visualizer.

## Starting a New Session

If you want to start a new session (File > New Session), all data from the current session is cleared, unless you save it as a metafile. When you start a new session, the graphics window is cleared.

## Obtaining Session Information

You can obtain information about your current session by selecting File > Session Information. The Session Information dialog box provides specific information about the graphics in your 3D scene and your workstation graphic capabilities. It provides a current count of the following components in your 3D scene:

- Graphic files
- Objects
- Polygons
- Average number of polygons for each object

In addition, the Session Information dialog box provides the following hardware-related details about your workstation:

- Z buffer size
- Graphic planes
- Maximum number of colors and lights enabled through your hardware
- Hardware Cutting Planes
- Amount of graphics memory used by your workstation

## Quitting EPD.Visualizer

You can quit EPD.Visualizer by selecting File > Quit to return to the operating system prompt.

Chapter 2

# Basic Features

This chapter provides an overview of the following features so that you can use EPD.Visualizer more effectively and increase your productivity:

- Display Controls
- Light Management
- View Processing
- Selecting Items
- Verification
- Item Editing
- Traversal
- Interference Checking
- Text
- Accessing Online Help

# Display Controls

This section describes the aesthetic changes you can make to control rendering and enhance the realistic effect of your 3D scene. The graphic display controls described in this section are found in the View menu.

## Rendering

EPD.Visualizer enables you to generate realistic looking 3D scenes with a combination of material properties, lighting, and several rendering methods.

How do you choose which rendering method to use? It depends on the type of 3D scene you want to create. In the rendering of part of a production process, for example, light, shadow, and reflection can provide important information. In the rendering of an underwater scene, too much light and reflection can be distracting.

Please note: Radiosity and ray tracing are very memory intensive, which limits their use on larger 3D scenes.

Rendering methods available in EPD.Visualizer include:

- Wire Frame — Displays all surface edges of the 3D scene as lines. It displays all lines in the model, whether or not those lines are visible to the viewer.
- Hidden Line — Displays a solid view of the 3D scene, with black polygon edges. The polygon faces appear equally bright because lights have no effect in the 3D scene with this option.
- Flat Shading — Displays a shaded solid view of the 3D scene, with the edges between the polygons visible and light settings incorporated.
- Smooth Shading — Displays a smooth shaded solid view of the 3D scene. Polygons that are defined as shells in the graphic files display so that you do not see the edges between them. Refer to Appendix B, “EPD.Visualizer Textual Graphic File” for a description of a shell and shell definitions.
- Radiosity — Displays the light that is reflected between diffusely reflecting surfaces in a 3D scene. Objects are illuminated by direct light sources, such as light bulbs, as well as light reflected from other objects in the scene. For example, when light shines on a wall, the wall radiates its light and color to objects close to it. Alternately, objects close to the wall radiate their light and color to the wall.

The quality of an image rendered with radiosity depends on lighting, light interpolation, and shadow and illumination sensitivity. EPD.Visualizer enables you to break up large surfaces to produce realistic shadows and illumination. You can control sensitivity to shadows and illumination by setting the overall sensitivity, local sensitivity, and boundary continuity.

Please note: Radiosity does not handle specular lighting and cannot display specular highlights, reflections, or refraction. Radiosity produces view independent images, as opposed to ray tracing, which produces transient, pixel based images.

- Ray Tracing — Displays reflections and refractions in the 3D scene. Ray tracing traces beams of light from the target back to the light source, taking into account reflections, mirroring, transparency, shadows, and refraction. It does not handle matte or diffusely reflecting surfaces. Ray tracing produces transient, pixel based images. To see a new view of a scene, you must generate the ray tracing operation again.

Please note: Because ray tracing is pixel based and uses a lot of computer resources, you might want to keep window sizes small when performing ray tracing operations.

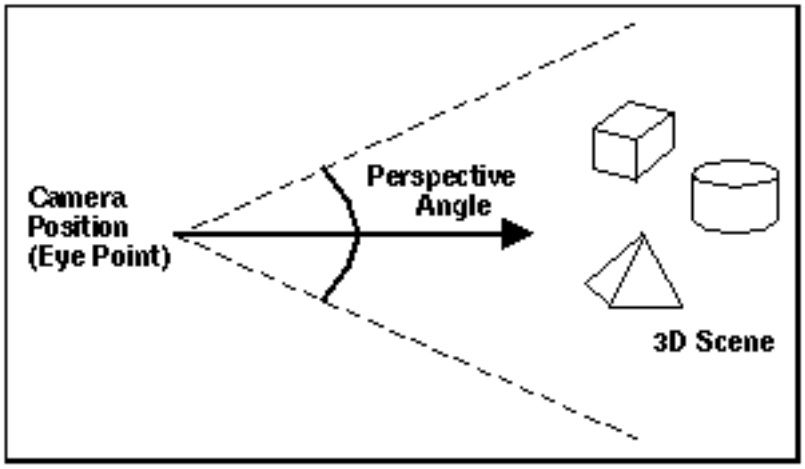
## Perspective Angle

You can increase or decrease your viewing area the same way you might see a scene differently through various camera lenses, such as a wide angle lens.

The Perspective option enables you to change the perspective view of the 3D scene. The value shown on the Perspective Angle slider scale represents the angle that exists from the camera position to the 3D scene. As you increase the perspective angle, the viewing area widens and the view of the scene changes accordingly.

Figure 2-1 shows how perspective is measured from the camera position to the 3D scene.

**Figure 2-1 Perspective Angle**



# Light Management

With EPD.Visualizer, you can define lights with independent direction and color. You can specify directional, point, and spot lights, as well as duplicate a light into a matrix of rows and columns. You can also manage the overall lighting environment by setting the ambient lighting and brightness controls.

You can incorporate realism and enhance the quality of your 3D scene through the appropriate application and management of light colors. The following features described in this section are found in the View menu:

- Specify type of light and color (Lighting > Editor)
- Set ambient lighting and overall brightness (Lighting > Environment)

## Directional Lighting

A directional light is infinitely far away from the objects it illuminates. Its direction is defined by a simple vector that starts from the light source position you specify and passes through the origin (0,0,0) of your 3D scene.

All light rays from a directional light are parallel and uniformly illuminate all surfaces in your 3D scene, regardless of location. Color and direction determine the light characteristics of directional lighting.

## Point Lighting

A point light is like a tiny light bulb. It occupies one point and radiates light equally in all directions. Color and position, not direction, determine the characteristics of point lighting.

## Spot Lighting

A spot light is a type of point light. You can specify a sharp-edged spot light, a variable-focus flashlight, or a fuzzy-edged flood light. Color, position, target, illumination angle, and edge sharpness attributes determine the characteristics of spot lighting.

## Matrix Duplication of Lights

EPD.Visualizer enables you to duplicate selected point, spot, rectangular, and circular lights in a matrix of rows and columns. This option is not available for directional lights.

## Ambient Lighting

Ambient light is a diffuse background lighting radiating from all directions. Using ambient lighting with directional lighting results in a softer atmosphere.

With ambient lighting, all planar surfaces in the 3D scene are illuminated. This enables you to examine objects that might be difficult to view in low lighting conditions.

## Environmental Brightness

You can control the overall intensity of lights in your 3D scene by setting the Environment Brightness slider scale. This scale adjusts all lights, including ambient lighting, ranging from Dim to Bright.

# View Processing

This section briefly describes views and the viewing capability of EPD.Visualizer. You can show and set the primary view, view all objects, save views and notes, and define and play back view sequences. The menu items described are located in View on the menu bar.

You can save and retrieve individual views with a specific camera orientation, including camera position, camera target and perspective angle. You can then organize these views into sequences for subsequent review or presentation.

In addition, you can annotate each view with descriptive information such as engineering data, design changes, IFC conflicts, or reminders. You can save any number of views, which you can later save with a metafile.

## Showing and Setting the Primary View

The primary view is a view that you can return to if you become disoriented while moving through your 3D scene (View > Primary View > Show). The initial primary view is defined when you add your first graphics file.

You can set a new primary view by selecting View > Primary View > Set.

## Viewing All Items

To display all items centered and within the limits of the graphics window, select View > View All.

## Saving Views and Notes

You can save views and notes for redisplay later by selecting View > View Editor. Then assign specific names to each view, and edit the notes later. After you save the views, you can organize them into a view sequence for play back.

## Defining and Playing Back View Sequences

To define a sequence of views to play back, select View > View Sequence > Editor. You designate previously-saved views as members of a view sequence. As you define a view sequence, you can display a view or review its associated note. You can also edit a view sequence to reorganize the views.

After you define the view sequence, select View > View Sequence > Playback. You can playback the view sequence in one of two methods:

- Interactive — Manually control when each view in the sequence is displayed, as well as read the associated note.
- Time Delay — Automatically display the entire view sequence with a delay between each view.

# Selecting Items

This section briefly describes building selection lists, setting selection masks, selecting and clearing items, and creating selection sets to help you delineate precise areas of interest. Selections made from EPD.Visualizer itself display messages in the status window that describe the selection type and show its name. Selections made through other tools display the message `Client command finished`.

## Building a Selection List

A selection list stores items that you select. After the items are added to the selection list, you can highlight them and subsequent selections by activating the Select > Highlight Selection toggle selection. Operations on objects available from other menus include Hide, Show, Translate, Rotate, and Delete. In addition, you can set up traversal by items, interference detection, or sectioning.

You can add individual items to the selection list with the Select tool. You can build a selection list that consists of Graphic Files, Selection Sets, Disciplines, and Groups by selecting Select > Build Selection Set. Disciplines and groups are attributes of items defined in the input graphic files. You can also select items that are hidden.

## Setting Selection Masks

Before you make selections with the Select tool, you can set a selection mask to specify graphics files, disciplines, groups, or items by selecting Select > Selection Masks. The selection mask default is graphics files.

After you make a selection in your 3D scene, the item, or all items within the discipline or group, are added to the selection list.

## Selecting and Deselecting Items

You can select items to add to the selection list with the Select tool. To select more than one item, press the `Shift` key and continue to select items with the Select tool. You can also select items by volume with the Volume Select tool.

You can clear the selection list either by selecting Select > Deselect All or by clicking in empty space in the graphics window. You can remove items from the selection list with the Deselect tool. The Deselect tool also clears items from the product structure tree in EPD.Connect.

## Defining Selection Sets

A selection set is a named collection of items that you can save and later add to the selection list. You define selection sets by adding the items to the selection list and then with Select > Selection Set Editor. You can save selection sets with a metafile.

After the selection set is defined, you can add it to the selection list in a single step with Select > Build Selection List. For example, if you define a selection set consisting of geometry that forms a forklift, you can easily add it to the selection list for item traversal.

# Verification

You can query EPD.Visualizer for item information with the verifying and measuring tools in the palette.

The following tools, located in the Tool palette, provide specific information about the 3D scene:

- Verify
- Measure Distance from Camera to Item
- Measure Distance Between Items

All verification data is calculated in working units (feet, miles, meters, etc.). You can define these units by selecting Options > Preferences.

## Verify

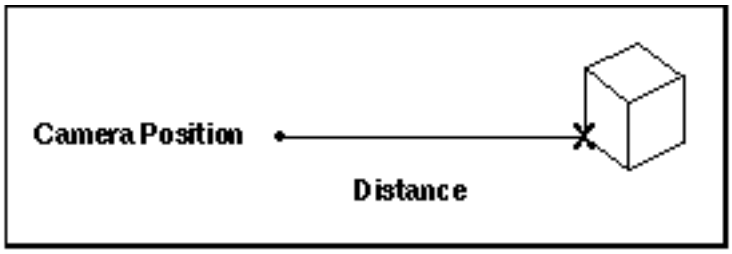
With the Verify tool, you can quickly and easily obtain detailed data about selected items.

After you select an item in your graphics window with the Verify tool, an Item Verification report displays information available from the graphics file. If the database is open, EPD.Visualizer also provides information from the external database.

## Measure Distance from Camera to Item

With the Measure Distance from Camera to Item tool, you can measure the distance from the camera position (your eye point) to a vertex point you select in the 3D scene. The following figure illustrates this concept:

**Figure 2-2 Measuring Distance from Camera to Item**



After you select the vertex of an item, the calculated distance from the camera to the item is displayed.

## Measure Distance Between Items or Graphics Files

The Measure Distance Between Items tool measures the distance from one vertex to another vertex. This tool is available in the Tool Palette as well as from the Tools menu (Tools > Measure Distance > between Items). After you select two vertex points in your 3D scene, the following measurements are displayed in the status window:

- The straight line distance between items
- The delta xyz distances

The straight line distance is calculated from vertex to vertex, and the delta xyz distances are expressed as the xyz component parts of the straight line distance.

The Tools menu includes a selection that allows you to display the minimum distance between two graphics files: Tools > Measure Distance > mini Graphics Files. This is useful when you are dealing with files that contain multiple items. The minimum distance is evaluated in one of the following manners:

- between vertices and other vertices
- between vertices and edges
- between vertices and facets

Two x-shaped markers and a line connecting them in the graphics window indicate where the minimum distance is measured. Distances between edges are not displayed, which can result in the display of inaccurate information.

## Measuring Distance in Sections

In addition to measuring distances for verification, you can also measure distances in sections. The tool available for doing so is available in two places, as follows:

- From the Tools menu (Tools > Measure Distance). The options starting with Sections Points apply to sections.
- From the Section window. The Section window appears when you enable sectioning from the Sectioning Control window. For more information, refer to Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning.”

# Item Editing

You can modify the items in your 3D scene in various ways with the EPD.Visualizer editing functions located in the Edit menu. These editing options include the following:

- Controlling the visibility or color of items
- Removing items from the 3D scene
- Translating or rotating items
- Traversing items, stepwise, along predefined paths
- Changing the surface material of planar objects

## Controlling Visibility

You can prepare uncluttered views by hiding the following object types with the Edit > Hide operation: Selected Items, References, or Unselected Items. References are explained in “Path Editor” on page 3-28.

You can restore the following object types to visibility with the Edit > Show operation: Selected Items or References. You can control the size of references revealed by the Edit > Show > References operation by specifications in the Path Router dialog box.

The Edit > Hide All operation displays a menu that enables you to hide all objects, paths, references, lights, or text. If you want to show the hidden items on the selection list, you can select Edit > Show. Edit > Show All displays a menu that enables you to restore visibility of hidden objects, paths, references, lights, or text. The size of references revealed by the Edit > Show All> References operation is set automatically.

## Deleting Items

To permanently delete items, add items to be deleted to the selection list and select Edit > Delete.

## Translating and Rotating Items

The move and rotate features enable you to experiment by making changes to your 3D scene, including checking for new interferences caused by your modifications. You can also simulate construction and assembly operations involving movement of items in the 3D scene. “Item Dynamics” on page 3-15 explains how to translate and rotate items.

You can change the position and orientation of items on the selection list by selecting Edit > Item Dynamics > Translate/Rotate or cause the items to traverse a predefined path by selecting Edit > Item Dynamics > Traversal.

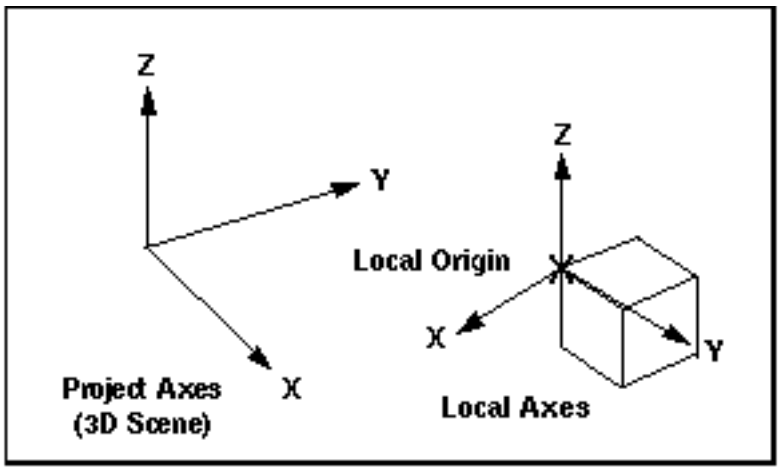
## Spatial Orientation

The possible directions of movement or axes of rotation for an item are the positive and negative axes of the global project or local coordinate systems.

The local axes coordinate system is attached to the items on the selection list and moves or rotates with the items. Initially, the local axes are parallel to the project axes with their local origin at the geometric center of the selection list.

You can move and rotate the local axes to any other position and orientation. The following figure shows an example of axes modification.

**Figure 2-3 Modified Local Coordinate System**



## Target of Translation and Rotation Operations

You can designate the item of your translate/rotate operation by choosing either the selection list or the local axes as the target.

If you choose the selection list as the target, you move the items in the selection list along with the local axes. If you choose the local axes as the target, you can change the relative position and orientation of the local axes with respect to the item geometry. This is useful for the following three reasons:

- Defines the direction in which to move the selection list
- Defines the axes about which to rotate the selection list
- Sets the orientation of the item when it traverses a path

The modified position of the local axes remains as long as the items remain on the selection list. After you modify the local axes of a selection list, you can save them as a selection set along with the items on the selection list with the Selection Set Editor. If the selection list was previously saved as a selection set, you must save it again under a new name in order to keep the modified axes.

## Translation

The Translate/Rotate option enables you to specify a distance on a slider scale and choose a direction along which to move the target. You can control the direction in which the selection list moves in a single translation by adjusting the orientation of the local coordinate system. You also have the option of moving the target in successive translations along different global coordinate directions.

Text is translated in the same way as other items; however, text cannot be rotated. If you select an object that has been tagged with text, the text is selected along with the geometric object. (You can tag an object with text either manually or automatically with the Object Tagging tool.)

## Rotation

The Translate/Rotate option enables you to specify an angle on a slider scale and choose an axis about which to rotate the target. You can rotate the selection list about one of the global axes or one of the local axes, depending on which one you select as the Active Coordinate System. In either case, the axis of rotation passes through the origin of the local coordinate system. Choosing a negative axis results in a clockwise rotation and choosing a positive axis results in a counterclockwise rotation about the selected axis.

## Item Traversal

The traversal option enables you to move the items on the selection list along predefined paths or Gpaths. Paths and Gpaths are segmented lines defined in textual graphic files (`.gaf` files). Gpath vertexes are called references and have names and orientation specifications that control the orientation of traversing items. “Defining Path and Gpaths” on page 2-19 summarizes the characteristics of paths and Gpaths. “Item Dynamics” on page 3-15 explains how to move items along a path or Gpath, and how to modify reference point or local axes.

When you perform item path traversal, you can define the following attributes:

- Specify a predefined path
- Define the step size and angle increment
- Set the orientation of the item, with respect to the path
- Select either interactive or timed stepping

All three orientation modes (Constant, Vertical, and To Path) position the item so the origin of its local coordinate system is placed at the start of the path. The local origin remains on the path as the item traverses each step. You can change the orientation and positioning of the item on a path in the following ways:

- Set the appropriate orientation mode in the Item Path Traversal dialog box
- Adjust the local axes of the item in the Translate/Rotate Selection dialog box

The orientation and position of an item or sectioning plane traversing a Gpath are defined in the Gpath, and you cannot change them with specifications in the Item Path Traversal dialog box.

An item traversing a path stops with its local origin at each path vertex and then rotates in steps of the angle increment until it is appropriately aligned to the next path segment. An object traversing a Gpath changes orientation continuously between references.

## Dynamic Interference Checking

If you turn on interference checking, the selection list is checked dynamically for interferences with all surrounding objects after each translation or rotation, or after each step during item traversal. An option allows the storage of positions that imply collisions with another item when an item traverses a path or Gpath, to allow you to reproduce the events. Refer to Chapter 5, “Interference Checking” for detailed information about interference checking.

## Item Color

EPD.Visualizer enables you to enhance the color of items in your 3D scene beyond the level provided in your original graphic files. By changing the colors of items, you can

- Draw attention to certain components
- Color code items according to particular classification (for example, construction status, assembly order, or engineering discipline)
- Increase the realism in appearance of geometric items

To modify the color of items, select the items and then select Edit > Visual Properties > Item Color. The Color Editor dialog box provides the following options:

- Select an existing color from the Colors list
- Select an existing color in your 3D scene
- Define a new color

You can define new colors by adjusting the Color Composition slider scales.

## Surface Materials

In addition to defining colors for objects in your 3D scene, you can define material properties for them. Material properties control the following features:

- Color of a surface
- How much light it reflects
- How transparent or opaque it is
- Color and intensity of its reflection
- Focus of the glare produced by the surface when light shines on it

These factors are primarily important when using ray tracing.

Surface material colors take precedence over an object's original color when you define a base color. If you define the base color of a copper surface material, for example, this color takes precedence over the original color of the object. To maintain the original color of the object, use the colorless feature.

To define material properties for surfaces, select the objects, then select Edit > Visual Properties > Surface Materials.

# Traversal

This section briefly describes paths and Gpaths, creating path and Gpath definitions, camera traversal, sectioning plane traversal, and item traversal. A path is a sequence of line segments consisting of any number of points, along which you can traverse objects, and a Gpath is similar to a path except that its points include orientations. Sectioning planes, objects, and the camera can traverse paths; sectioning planes and objects can traverse Gpaths. You can add any number of paths or Gpaths to your 3D scene.

## Defining Path and Gpaths

You can define paths or Gpaths by creating graphic files or by using the Path Editor dialog box. A path definition contains the units, the path name and color, and the coordinates of the path. A Gpath definition includes references, and each Gpath reference has a location, rotations around the global axes, and an optional name. After you create the graphic files that define paths and Gpaths, you can select the paths and Gpaths to be displayed as lines within the 3D scene and to guide traversal. Refer to Appendix B, “EPD.Visualizer Textual Graphic File” for information on creating or editing a textual graphic file containing path definitions or Gpath definitions. The “Path Editor” on page 3-28 explains how to define paths and Gpaths with dialog boxes.

## Camera Traversal

You can traverse the camera along a selected path in the 3D scene by selecting View > Camera Traversal. The camera is focused in any of the following ways:

- Horizontal lock
- Along the path
- Fixed point
- Fixed direction

You can adjust the step distance or incremental turn angle along a path.

Camera Traversal is helpful for evaluating every angle as you walk through the 3D scene. For a more realistic view when traversing the camera along a path, turn on Double Buffering in the Rendering Control dialog box.

## Item and Section Plane Traversal

You can traverse items you added to the selection list along a selected path by selecting Edit > Item Dynamics > Path Traversal. If you turn on interference checking, all interfering items are highlighted. Chapter 8, “Moving a Plane” explains the movement of a sectioning plane on a path. “Sectioning the Scene from Multiple References” on page 7-14 explains how you can simultaneously see the all sections resulting from sectioning a scene with a plane that is oriented by some or all the references in a Gpath.

Path traversal simulates the movement of items in the 3D scene to check clearances, accessibility, and interferences. For a more realistic view when traversing along a path, turn on Double Buffering in the Rendering Control dialog box.

# Interference Checking

With interference checking, you can perform spatial studies to detect potential design flaws early in the engineering process before construction or fabrication begins. Interference checking features are easily accessed through menu options. Specific capabilities include:

- Static and dynamic interference checking
- Coloring and highlighting colliding components
- Reviewing the results
- Generating reports of the results

Static and dynamic interference checking are described in the following sections. Refer to Chapter 5, “Interference Checking” for additional information about interference checking.

## Static Interference Checking

Static interference checking allows you to manually check for interferences between stationary components belonging to the current selection and the surrounding three-dimensional scene. You can perform a static interference check by selecting components in your scene and performing interference checking as described in “Setting Up an Interference Checking Session” on page 5-4.

## Dynamic Interference Checking

Dynamic interference checking allows you to automatically check for interferences during item editing and item traversal. During item editing, interferences are automatically detected and highlighted for each translation or rotation operation. During item traversal, interferences are detected and highlighted at each step position along the traversal path. All positions implying interference are recorded in a new Gpath so that you can reproduce them. You can specify that the first interference stops the item traversal, and whether or not the traversing item remains at its last position. You can enable dynamic interference checking as described in “Turning On or Off Item Dynamics” on page 5-4.

# Text

This section briefly describes defining text properties; creating, editing, and deleting text; and attaching tags to objects, either manually or automatically.

## Defining Text Properties

Text properties are specific settings consisting of font, font type (variable or fixed), text alignment, font size, text color, and spacing between characters. You can define the text properties by selecting Edit > Text Properties before you enter text or attach tags. Otherwise, you can use the default text properties.

After you define the text properties, any text you add in the graphics window adopts these properties. You can also change the properties of existing text with the Select tool to select the text. After you change the properties, the selected text displays the new properties.

## Creating, Editing, and Deleting Text

You can create text anywhere in the graphics window with the Text tool. You can also edit existing text after you select it with the Text tool. You can delete text by selecting it with the Select tool and then selecting Edit > Delete. You enter text in the Text Editor, which also displays the xyz coordinates of the text.

## Attaching Text or Tags to Objects

You can manually attach text to objects, or you can automatically attach tags from the database to objects. If you select

- Database > Tagging > Automatic, after you open the database, predefined tags are attached to an object that you selected with the Object Tagging tool
- Database > Tagging > Manual, the text you enter is attached to an object that you selected with the Object Tagging tool

# Accessing Online Help

This section briefly describes using online help to obtain immediate assistance while you are using EPD.Visualizer.

You can learn how to use the context-sensitive help by selecting Help > On Help. You can access context-sensitive help by selecting Help > On Context, which provides online help for each menu bar name and each button, tool, and slider scale on the palettes.

In addition, you can obtain quick reference help as you point to the tools, buttons, or slider scales. A brief description of each one is displayed in the prompt window. You can also obtain online help for each dialog box by clicking the Help button.

For general how-to information on using the interface, select Help > On General. To access the EPD.Visualizer glossary, select Help > On Glossary. For information on environment variables, select Help > On Environment.

Refer to “EPD.Visualizer Main Menus” on page 3-2 in for additional information about online help.

Chapter 3

# Using EPD.Visualizer Menus

This chapter describes how to use each menu in the order they appear on the menu bar. For information on menu basics, refer to Chapter 1, “Introduction to EPD.Visualizer.”

- EPD.Visualizer Main Menus
- File Menu
- Edit Menu
- View Menu
- Options Menu
- Tools Menu
- Select Menu
- Database Menu
- Interference Checking Menu
- Sectioning Menu
- Online Help Menu

# EPD.Visualizer Main Menus

The following figure shows the menu bar located at the top of the EPD.Visualizer window.

The following briefly describes the menus on the menu bar:

- File — Manage graphic files and metafiles, start and stop sessions, and print screens
- Edit — Edit items and text properties and manage the display of items
- View — Show primary view, save and playback views, traverse the camera along paths, and manage lights, rendering and display controls
- Options — Customize working environment
- Tools — Use Tool Palette functions and Interact with the product structure tree
- Select — Manage selection of items
- Database — Access a database and set object tagging
- Interference — Perform interference checking, review the interferences, and generate reports
- Sectioning — Scan an assembly by defining and moving a plane
- Help — Access online help

# File Menu

By selecting File on the menu bar, you can access the following menu items:

- New Session — Start a new session
- Open Metafile — Open a metafile
- Add Graphics — Add textual or binary graphic files
- Remove Graphics — Remove graphic files
- Save Metafile — Save changes to an open metafile
- Save Metafile As — Save a new metafile
- Graphics Output — Capture view in graphics window to a raster or PostScript file
- Session Information — Obtain session information
- Quit — Quit EPD.Visualizer

After you start EPD.Visualizer, you can either add one or more graphic files or open a metafile.

## Starting a New Session

When you start a new session, all data from the current session is cleared unless you save it in a metafile. To start a new session, perform the following steps:

1. Select File > New Session. If you have not already saved your current session in a metafile, you are prompted as follows:

```
Current session has not been saved in a metafile!
Do you still want to start a new session?
```

2. If you want to save a metafile, click No and select File > Save Metafile As. Then select File > New Session again, and the graphics window is cleared.

   If you do not want to save the current metafile, click Yes and the graphics window is cleared.

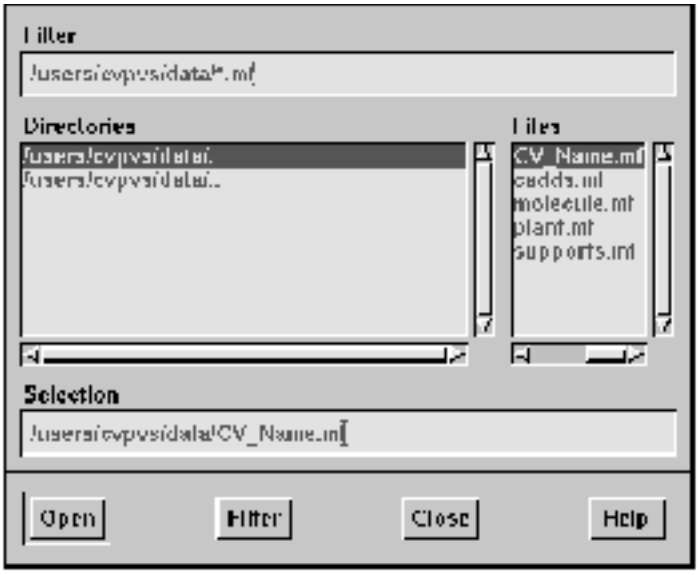
3. You can now add new graphic files or open a metafile.

# Opening a Metafile

A metafile is an ASCII file that contains a complete description of an EPD.Visualizer session. This file includes all geometric objects, views, view sequences, lights, materials, paths, color definitions, annotation, and control settings, such as working units and slider scale settings.

You can only open one metafile at a time. When a metafile is open the title bar displays the path name of the file. To open a metafile, perform the following steps:

1. Select File > Open Metafile to display the Open Metafile dialog box, as shown in the following figure.




2. Double click a metafile name in the Files list, or enter a metafile name in the Selection box and click Open.

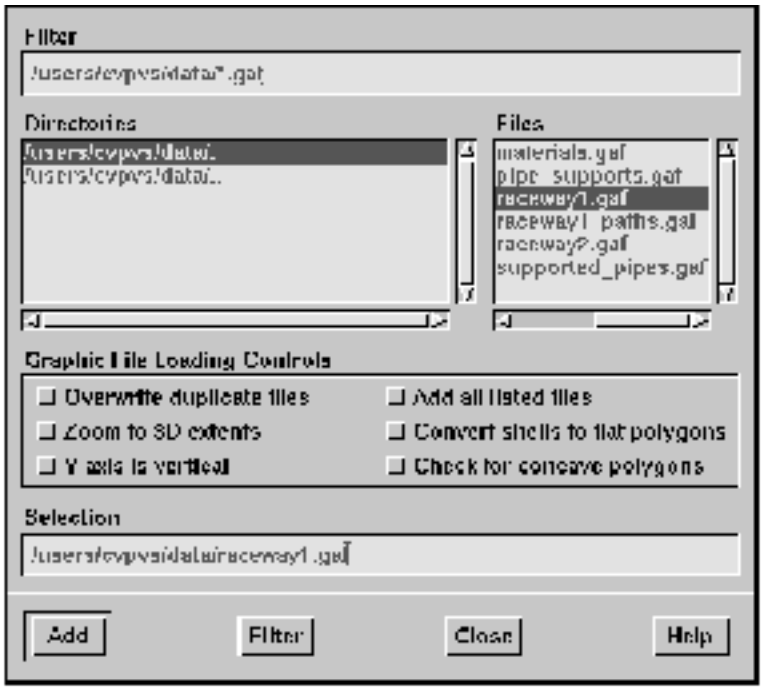
   Refer to the “File Selection Dialog Box” on page 1-17 for more information on using the filter option.

# Adding Textual or Binary Graphics Files

You can save numerous textual or binary graphic files in one metafile. Textual graphic files are named with a `.gaf` extension. Binary files are named with a `.gbf` extension.

To add a graphic file to a metafile, perform the following steps:

1. Select File > Add Graphics > Textual or Add Graphics > Binary to display the Graphic File Selection dialog box as shown in the following figure.




2. Set the following Graphic File Loading Controls options as necessary:
   - Overwrite Duplicate Files — Add a graphic file that is currently displayed in the 3D scene, overwriting the one already displayed.
   - Zoom to 3D Extents — Add several graphic files and display them in full view within the limits of the coordinates of the objects.
   - Y Axis is Vertical — In EPD.Visualizer, the vertical orientation is based on the z-up coordinate system. If your graphic files were created based on a y-up coordinate system, select this option. EPD.Visualizer automatically converts y-up graphic files to z-up.

- Add All Listed Files — You can add all of the files in the listing by selecting this option. A status box displays the following information:

  — Total number of files to be added

  — The number of the file being processed

  — Number of files added

  — Number of files with errors

- Convert Shells to Flat Polygons — To increase the rendering speed of the 3D scene, select this option to display all shell polygons as flat polygons.
- Check for concave polygons — Check each flat polygon for concavity and set proper flags to ensure the correct display. This option is not needed for EPD.Visualizer running on HP, IBM, and Digital. On the SGI and UltraSPARC, this option should be set when the input data contains concave polygons. By default, this option is set for the SGI, but not set for the HP, IBM, Digital, and Sun. When this option is selected and the environment variable PVS_CONCAVE_PERFORM is set to a non-null value, concave polygons are converted into a set of triangles. After conversion, the image is stored in RAM, which improves performance, but may be limited by RAM limitations. If the image is not converted, updating the screen may require more time, but the image is not stored in RAM.

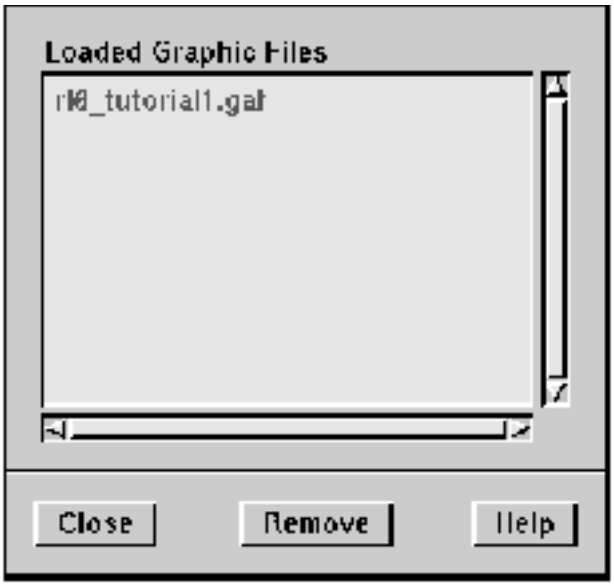
3. Double click a file name in the Files list, or enter a file name in the Selection box and click Add.

   Refer to the “File Selection Dialog Box” on page 1-17 for more information on using the filter option.

# Removing Graphic Files

You can remove textual or binary graphic files from the metafile. This action removes all geometry associated with the removed files. To remove a graphic file, perform the following actions:

1. Select File > Remove Graphics. The following figure shows the Graphic File Removal dialog box.



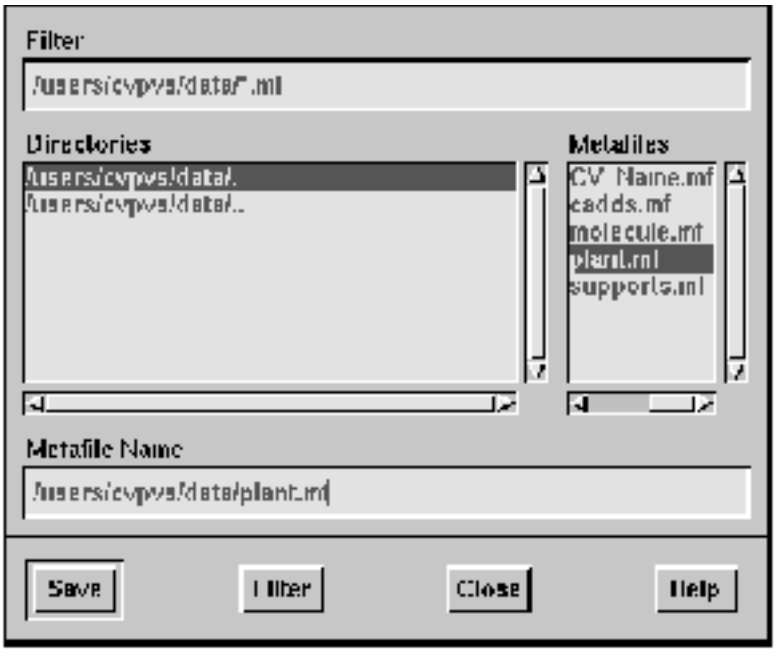

2. Click the file name(s) from the list of Loaded Graphic Files.

   To select multiple file names, hold down the `Control` key and click the file names you want to select.

3. Click Remove and then Close.

## Saving a Metafile

When a metafile is open, the title bar displays the metafile path name. To save changes to a new metafile, select File > Save Metafile As to display the Save Metafile dialog box as shown in the following figure.




Type the name of the file in the Metafile Name field, and click Save. The Save Metafile Options Selection box is displayed next.

By default, EPD.Visualizer saves metafiles with all geometry, materials and path information. If you want to save the entire metafile without any of this information, select the information categories you do not want saved and click OK.

To save recent changes to an existing metafile, select File > Save Metafile to display the Save Metafile Options Selection box.

When you choose to save a metafile without a particular type of information, the entire metafile is saved without the information, not just the information added since the previous save.

Refer to the “File Selection Dialog Box” on page 1-17 for more information on using the filter option.

# Graphic Output

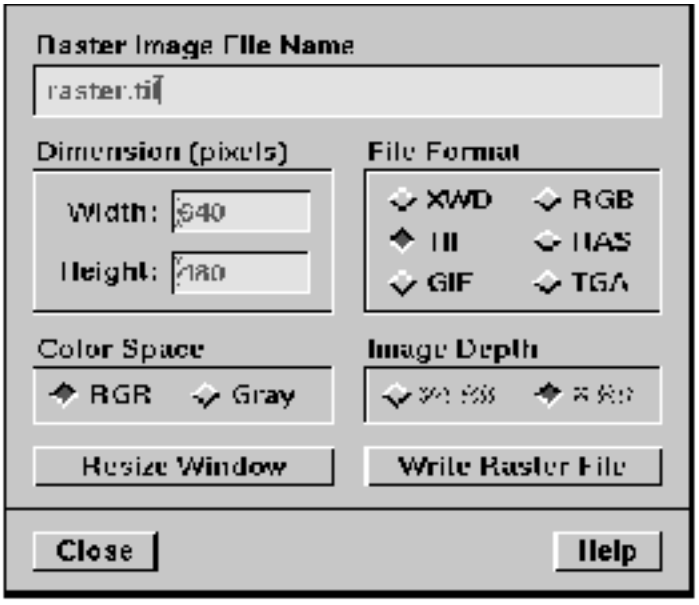
## Capturing a Raster Image

You can capture a raster image of a view displayed in the EPD.Visualizer graphics window for printing or viewing outside of EPD.Visualizer. To use this feature, the EPD.Visualizer `pvs_write_raster` program must be available and executable on your system. This program is distributed with EPD.Visualizer and is installed on the `${VIS_HOME}/bin` directory.

Please note: On Sun platforms running `xgl`, do not allow any window or menu to occlude the EPD.Visualizer window while performing the capture operation. Otherwise, the output raster file will not be created properly.

To capture a raster image, perform the following steps:

1. Select File > Graphics Output > Raster Image. The Raster Image Capture dialog box displays.




2. In the Raster Image File Name field, enter the file name with or without a directory path.
3. Set the following options as necessary:
   a. Dimension
      — Width: pixels
      — Height: pixels

   After entering the new dimension, click Resize Window.

**b.** File Format

— XWD: X11 windows dump format

— TIF: Tagged Image File

— GIF: CompuServe Graphics Image File

— RGB: Silicon Graphics Image format

— RAS: Sun Microsystems Raster File

— TGA: TruevisionTarga Image File

**c.** Color Space

— RGB: Full color image

— Gray: Gray scale image

**d.** Image Depth

— 24 bit image: Use the complete color spectrum

— 8 bit image: Average some color settings

All image file formats support 24- and 8-bit image depths except for GIF, which is limited to 8 bit.

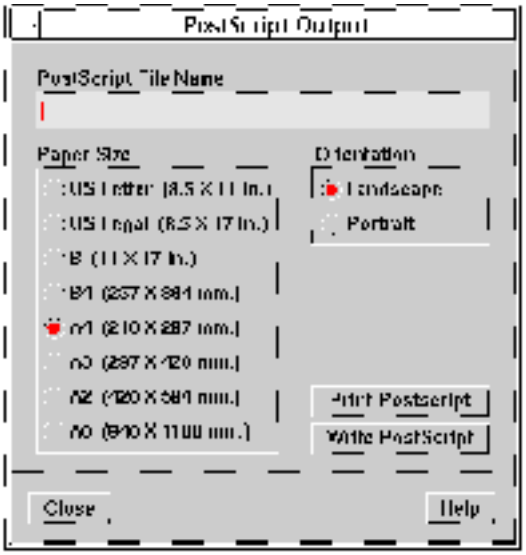
4. To save the raster image, click Write File.
5. To view the raster image, at the operating system prompt, enter the following command:

```
display filename
```

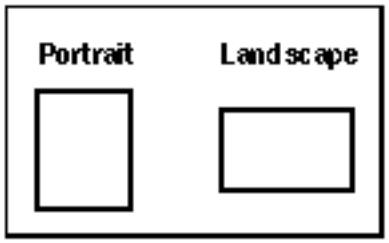
## PostScript Graphics Output

You can capture an image of a view displayed in the EPD.Visualizer graphics window in PostScript, format the image, and print it. To do this, perform the following steps:

1. Select File > Graphics Output > PostScript to display the PostScript Output dialog box.




2. In the PostScript File Name field, enter the file name with or without a directory path. You can overwrite an existing file name by entering its name.
3. Select the paper size. The image is scaled to the paper size you select.
4. Select the orientation of the page. The options are shown below:




5. To save the file, click Write PostScript and then Close. You can print the file from the operating system.

The file that prints is a temporary file (`/usr/tmp/sectionoutput.tmp`) that is automatically deleted after printing. By default the `lpr` command prints the file, but you can specify a different command for PostScript files by defining the `PVS_PRINT_COMMAND` environment variable. You give the variable the value of an executable file such as a binary or script; no additional argument for the command

is allowed. For example, the following line changes the default print command for PostScript files to `lp`:

```
setenv PVS_PRINT_COMMAND /usr/bin/lp
```

To print the contents of the section window you perform the following actions.

1. Set the options:

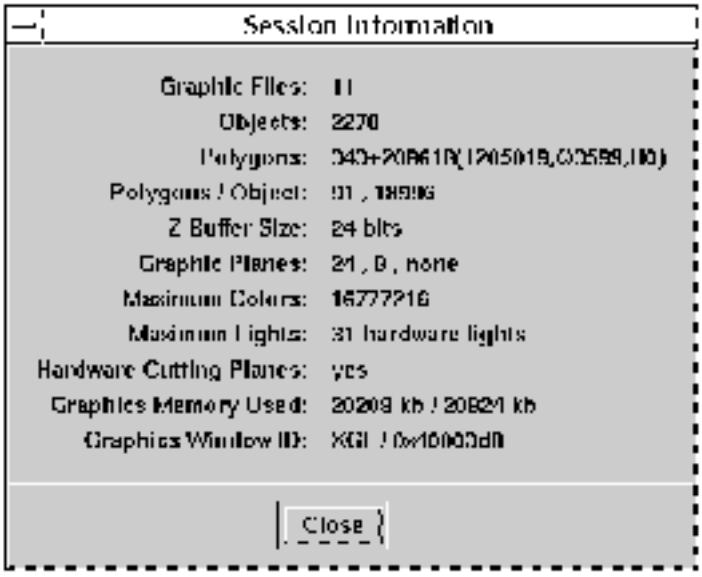
   Paper Size — If one of the standard paper formats is selected, the contents of the section window will be automatically scaled to fit the paper size with adjustment for the orientation.

   Orientation — Select Landscape or Portrait.

2. Press the Print PostScript button.

## Obtaining Session Information

To obtain specific information about the graphics in your 3D scene and your workstation graphic capabilities, select File > Session Information to display the Session Information window shown in the following figure.



Session information consists of a current count of the following:

- Graphic files
- Objects
- Polygons — Displays five counts:
  - Number of flat polygons
  - Number of faces defined in shells
  - For faces defined in shells, T precedes the number of triangles
  - For faces defined in shells, Q precedes the number of quadrangles
  - For faces defined in shells, H precedes the number of holes
- Polygons / Object — Displays two counts:
  - Average number of faces, including flat polygons, by object
  - Average number of faces, including flat polygons, by graphics file

In addition, the following hardware-related information displays:

- Z buffer size
- Graphic planes
- Maximum number of colors and lights allowed with your hardware
- Hardware Cutting Planes
- Amount of graphics memory used by your workstation
- X11 window ID of the EPD.Visualizer graphics window

## Quitting EPD.Visualizer

1. Select File > Quit and you are prompted:

   ```
   Are you sure you want to quit?
   ```

2. Click Yes to return to the operating system prompt, or click No to continue with your current session.

# Edit Menu

By selecting Edit on the menu bar, you can access the following menu items:

- Show — Show hidden items or references of selected Gpaths
- Hide — Hide selected items, unselected items, or references of selected Gpaths
- Show All — Show all objects, paths, Gpaths, references attached to Gpaths, lights, or text
- Hide All — Hide all objects, paths, Gpaths, references attached to Gpaths, lights, or text
- Item Dynamics — Translate, rotate, or move items along a path; modify reference point/local axes; or perform multiple sectioning from Gpath references
- Path Editor — Generate path segments within a 3D scene
- Visual Properties — Define color and material properties for items
- Text Properties — Define and edit text properties
- Delete — Delete items

## Show

To show hidden items on the selection list, select Edit > Show > Selected Items. To show the references of Gpaths on the selection list, select Edit > Show > References. You can set the size of the displayed references in the Path Router dialog box.

## Hide

You can hide items or references of selected Gpaths. To hide items on the selection list, select Edit > Hide. The items remain hidden until you select Edit > Show or Edit > Show All. After you hide items, the selection list is cleared.

To hide the items selected and display only the unselected items, select Edit > Hide > Selected Items. To hide all the unselected items and display only the selected items, select Edit > Hide > Unselected Items.

To hide selected items, use the following steps:

1. Highlight the items you want hidden.

   To select multiple items in a 3D scene, hold down `Shift`, then click each item.
2. Select Edit > Hide > Selected Items.

The selected items are hidden in the 3D scene.

To display only the selected items, use the following steps:

1. Select the items you want displayed
2. Select Edit > Hide > Unselected Items.

The selected items are displayed and the remainder of the scene is hidden.

To hide the references of one or more Gpaths, use the following steps:

1. Select the Gpaths whose references you wish to hide.
2. Select Edit > Hide > References.

## Show All

To redisplay hidden items, select Edit > Show All. The Show All menu appears. You can choose to show all objects, lights, paths, Gpaths, references attached to Gpaths, or text. References appear in a size that is appropriate for the 3D scene: 10% of its maximum visible measurement.

## Hide All

To hide groups of items, select Edit > Hide All. The Hide All menu appears. You can choose to hide all objects, lights, paths, Gpaths, references attached to Gpaths, or text.

Any new Gpath created by reading a `.gaf` file or using the Path Editor dialog box appears with references in the graphics window, regardless of the current path visibility setting.

## Item Dynamics

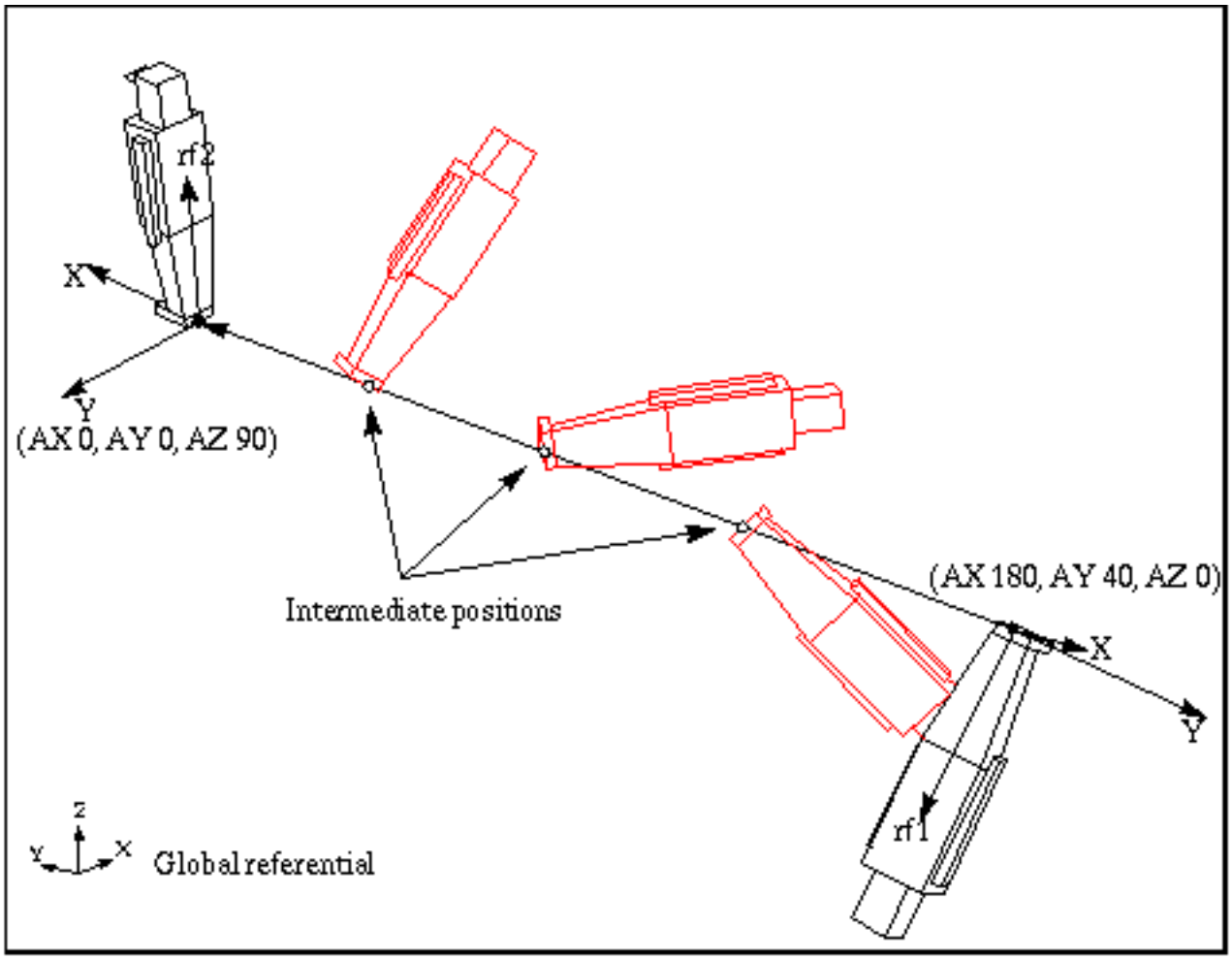
To translate, rotate, or move items along a path or Gpath, and to modify reference point or local axes, select Edit > Item Dynamics.

## Path Traversal

This option enables you to move objects along predefined paths and Gpaths through the 3D scene. The characteristics and control of path traversal and Gpath traversal differ. For an object traversing a path, you can select an orientation for the object relative to the path. For an object traversing a Gpath, you can specify the orientation for the object relative to the reference attached to each vertex.

The orientation of an object traversing a path remains constant between vertices, and if the orientation changes, the change occurs at a vertex. The orientation of an object traversing a Gpath changes continuously between vertices. At each point on a Gpath segment, the orientation is a linear combination of the orientations defined at each end of the segment. The linear combination is made on each rotation angle, and not on the referentials themselves.

In the following example, an object is moving from a reference `rf1` defined by the three angles (AX 180, AY 40, AZ 0) to `rf2` (AX 0, AY 0, AZ 90). At the middle point, the angles defining the position of the object will be (AX 90, AY 20, AZ 45). If you are expecting a different orientation at this place, you must use the Path Editor to create an intermediate vertex with the correct orientation.




You can define any number of paths or Gpaths in textual graphic files. After you add these textual graphic files, the paths and Gpaths appear as lines within the 3D scene. Gpaths appear with visible references. Refer to Appendix B, “EPD.Visualizer Textual Graphic File” for information on creating or editing a textual graphic file containing path definitions. For information about moving a sectioning plane along a path, refer to Chapter 8, “Moving a Plane.”

To execute item traversal controlled by defaults, and to configure item traversal if required, perform the following steps:

1. Add graphic file(s) with path definitions.
2. Select item(s) to traverse the path.
3. Select Edit > Item Dynamics > Path Traversal to display the Object Path Traversal dialog box. The following figure shows on the left the Object Path Traversal dialog box. The following figure shows on the right the Object Path Traversal Options dialog box which you display by pressing the Set Options button in the Object Path Traversal dialog box.

   You select a path or Gpath by clicking it in the Path Names scrolling list in the Object Path Traversal dialog box or by pressing the Select Path button and selecting a path in the graphics window. The vertices of the path or Gpath appear in the References/Points scroll list.

   The three following explanations describe the ways to accomplish traversal immediately, relying on default settings. If default settings are not appropriate, you need to configure the traversal in the Object Path Traversal Options dialog box as detailed in the next step:

   - The scrolling list in the Object Path Traversal dialog box displays all the vertices or references in the selected path or Gpath. Clicking on a list item moves the object directly to it. Buttons beside the scrolling list allow you to make list selections to which the object immediately moves.

     — First: The object moves to the position described by the first list item.

     — Prev: The object moves to the position described by the list item preceding the currently selected list item.

     — Next: The object moves to the position described by the list item following the currently selected list item.

     — End: The object moves to the position described by the last list item.

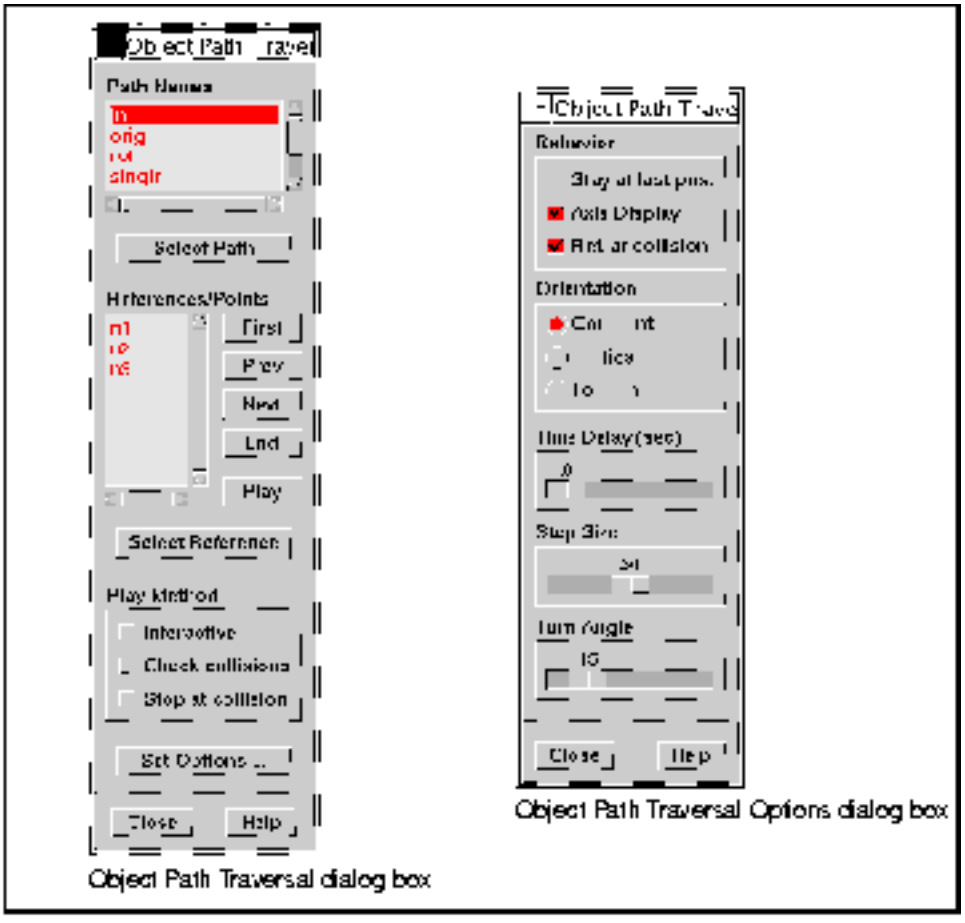
   - The scrolling list in the Object Path Traversal dialog box displays all the vertices or references in the selected path or Gpath. Push the Play button to make the selection traverse the path or Gpath automatically. The Play Method area contains buttons that control the traversal initiated by the Play button:

     — Interactive: Manually step along the path. If this method is not selected, the Time Delay method described in the next step is used.

     — Check Collisions: Activates Dynamic Interference Checking. During item traversal, interferences are detected and highlighted at each step position along the traversal path. “Setting Up an Interference Checking Session” on page 5-4 describes this feature. This button is automatically activated in the Stop at Collisions mode.

— Stop at Collisions: The first collision stops the traversal. This button is inactive if the Interactive button is selected. Activating the Stop at Collisions button automatically activates the Check Collisions button. Deactivating the Stop at Collisions button automatically deactivates the Stop at Collisions button.

- You can move an object directly to a graphically selected reference. Select a Gpath reference by pressing the Select Reference button and selecting a reference in the graphics window. When you make the selection, the Gpath is selected, the selected object moves to the reference directly without traversing intermediate points, and the reference is selected in the References/Points scroll list. A reference must be visible for you to select it. You make references visible by selecting Edit > Show > References or Edit > Show All > References. The Select Reference button is irrelevant to the two preceding methods and remains active after you have selected a reference.



4. Specify settings in the Object Path Traversal Options dialog box if required:

   - Behavior

     — Stay at Last Pos.: Selecting this makes the object stay at the last position when motion ends. If it is not selected the object returns to its original position.

     — Axis Display: Selecting this displays the local axes of the object during motion. If it is not selected the local axes are not displayed. You cannot switch this setting during path traversal as the button is inactive when motion starts.

     — Ref. at Collision: Selecting this creates a Gpath from the positions where collisions occur during object motion. It is not selected, the Gpath is not created. The Gpath is named collisions, and a new collisions Gpath replaces a preexisting one. The collisions Gpath displays in the Object Path Traversal dialog box and is selectable like other Gpaths. Traversal of the collisions Gpath does not cause it to be replaced. The default color of the collisions Gpath is white, but you can change it in the Path Editor for the current session.

   - Orientation

     — Constant: Maintain the original orientation of the item(s) along the entire path. This is not available for Gpaths.

     — Vertical: Rotate the item(s) horizontally to follow the path but always maintain the up direction (local Z axis direction). This is not available for Gpaths.

     — To Path: Rotate the item(s) to keep the forward direction (local X axis direction) aligned to each path segment and the up direction (local Z axis direction) in a vertical plane. This is not available for Gpaths.

   - Time Delay

     Automatically traverse the path with the Time Delay slider scale to set the number of seconds to delay between each step. If the Movie Recorder is turned on while with this method, movie images are generated for the traversal sequences. By default the units are seconds and tenths of seconds.

   - Step Size

     Set the step size the item(s) take along the path, with the Step Size slider scale. A zero value causes the item(s) to take one step for each path segment.

   - Turn Angle

     Set the angle by which the item(s) incrementally turn when reaching the corners along a path, with the Turn Angle slider scale. A zero value causes the item(s) to turn each path corner in one step.

To verify the name of a path, pick up the Select Path tool in the Object Path Traversal dialog box and select the path in the 3D scene. The name is highlighted in the Path Name list.

To check the orientation of the selected items click Start Position. The items appear at the beginning of the selected path with respect to the path direction.

## Translate/Rotate

With this option, you can perform the following operations:

- Move or rotate items on the selection list
- Move or rotate the local axes associated with the selection list

Translate enables you to move the selection list or axes while maintaining their current orientation. If you translate the selection list, the local axes move along with it.

Rotate enables you to rotate the selection list or its local axes about selected axes. The local axes rotate in the local coordinate system.

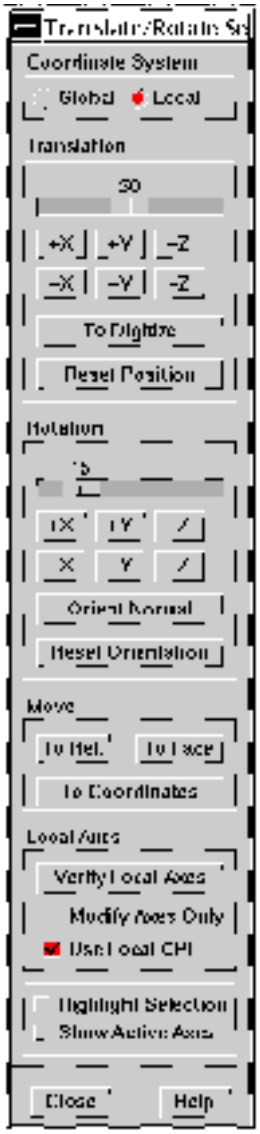
The local or global axes define the orientation of selected items. The local axes also determine how items on the selection list align with a path during item traversal. The default local axes of the selection list are parallel to the global axes (the XYZ coordinate system of your 3D scene). For further information refer to“Spatial Orientation” on page 2-14.

The default local origin is located at the geometric center of the selection list, but you can modify this by selecting the Use Local CPL button in the Translate/Rotate Selection List dialog box, which is shown in the following figure. Setting the environment variable PVS_LOCAL_CPL to any value allows you to start EPD.Visualizer with the Local CPL button enabled. Selecting the Use Local CPL button has the following effects:

- The local origin is at the virtual point X0 Y0 Z0 of the last selected graphic file.
- The X, Y, and Z axes are respectively defined in the EPD.Visualizer textual graphic file as `(1,0,0)`, `(0,1,0)`, and `(0,0,1)`.
- EPD.Connect is able to load objects with a translation and rotation of the graphic file. The local origin reflects the transformations applied to objects and accounts for the translations and rotations that occur when an object is loaded from EPD.Connect.
- Due to the three preceding effects, selecting the Use Local CPL button allows the reproduction of a local origin corresponding to the TOP Cplane used in creating the object in a CAD/CAM system.

Before you start Translate/Rotate, perform the following steps:

1. Add item(s) to translate or rotate to the selection list.
2. Select Edit > Item Dynamics > Translate/Rotate. The Translate/Rotate Selection List dialog box displays as shown in the following figure.




3. Set the following options as necessary:
   - Active Coordinate System
     — Global: Translate/rotate with respect to the global axes
     — Local: Translate/rotate with respect to the local axes

To translate with the Distance slider scale, use the following steps:

1. Set the unit of translation on the Distance slider scale.
2. Click the appropriate axis button:
   - Positive (+) X, Y, or Z moves the target in the positive direction along the selected axis.
   - Negative (-) X, Y, or Z moves the target in the negative direction along the selected axis.

   For example, if you want to move the target 5 units in the negative X direction, set the scale to 5 and click -X.

To translate with the To Digitize tool, use the following steps:

1. Pick up the To Digitize tool and move the pointer into the graphics window.

   If you are translating item(s) on the selection list, you are prompted with the following:

   ```
   Specify Selection List Location:
   Digitize new location for the selected items...
   ```

   If you are translating the local axes, you are prompted with the following:

   ```
   Specify Axes Location:
   Digitize new location for the axes...
   ```

2. After you digitize a new location, a digitize marker snaps to the nearest vertex.

   If you are translating the local axes, the local origin of the axes moves to this vertex. If you are translating the selection list, the axes and the selection list translate to the selected vertex.

3. Click To Digitize or select any other tool to deactivate the tool.

To rotate the selection or the local axes, use the following steps. If you select Local Axes as your target source, you must select Local as your coordinate system or the following warning displays:

```
You can rotate the local axes only in the local coordinate system.
```

1. Set the Angle slider scale to specify either of the following depending on your target source:
   - Angle by which the selection list rotates about a chosen axis
   - Angle by which the local axes of the selection list rotate about the local coordinate system

2. Click the appropriate axis button:
   - Positive (+) X, Y, or Z rotates in a counterclockwise direction about the selected axis
   - Negative (-) X, Y, or Z rotates in a clockwise direction about the selected axis

   For example, if you want to rotate 10 degrees in a clockwise direction about the X axis, set the scale to 10 and click -X.

To show active axes, use the following steps:

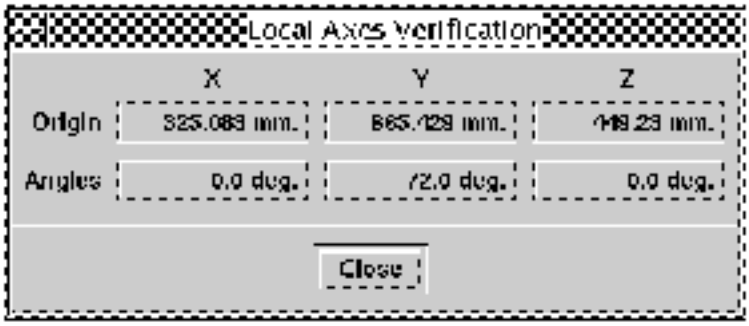
1. Select the axes you want to display (Global or Local) in the Coordinate System area.
2. Click Show Active Axes and the selected axes are displayed in the 3D scene.

To show the selection list click Highlight Selection and the selection list is highlighted in the 3D scene.

To verify Local Origin and Axes click Verify Local Axes in the Translate/Rotate Selection List dialog box. The Local Axes Verification dialog box displays the following information:

- Coordinates of the local origin in the global coordinate system
- Angles that would orient a reference in a Gpath like the current coordinate system

The information in the Local Axes Verification dialog box updates during motion, except motion initiated from the Object Path traversal dialog box. The Local Axes Verification dialog box updates after motion initiated from the Object Path traversal dialog box.




The Orient Normal button allows you to orient a selection so that its Z axis is parallel to a normal without moving the selection's origin. You press the Orient Normal button and select a surface or a reference.

With the buttons in the Move area, you can move selections in the following ways:

- Moving selection to a reference when the Modify Local Axes in the Local Axes area is not selected.

  You select the To Ref. button and then make a selection in the graphics window. The selection gets both the position and orientation of the reference.

  If the tool finds no reference close enough, it looks for a surface instead. If the tool accepts a surface, the selection moves to the nearest edge of the surface and the normal provides its orientation. The orientation and position are stored in a temporary reference. The value of the environment variable PVS_NORMAL_AXIS specifies which axis of the temporary reference is placed along the normal. PVS_NORMAL_AXIS has the default value Z; the other two axes receive arbitrary positions.

- Moving only local axes of selection relative to a reference, when the Modify Local Axes in the Local Axes area is selected.

  You select the To Ref. button and then make a selection in the graphics window.

- Moving the selection to a surface when the Modify Local Axes in the Local Axes area is not selected.

  You select the To Face button and then make a selection in the graphics window. The selection moves to the nearest edge of the surface and the normal provides is the basis for its orientation. The position and orientation are stored in a temporary reference. The value of the environment variable PVS_NORMAL_AXIS shows which axis of the temporary reference is placed along the normal. PVS_NORMAL_AXIS has the default value Z; the other two axes receive arbitrary positions.

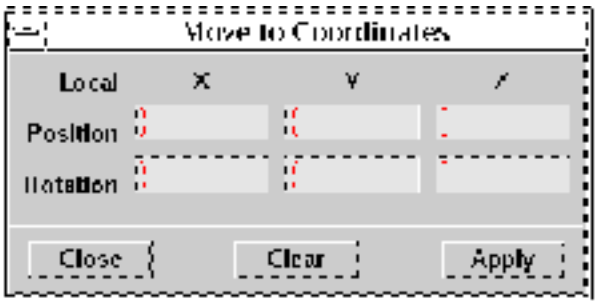
  If the tool finds no surface close enough, it looks for a reference instead, and the selection gets both the position and orientation of the reference.

- Moving only local axes of selection relative to a surface, when the Modify Local Axes in the Local Axes area is selected.

  You select the To Face button and then make a selection in the graphics window.

- Moving the selection to explicit coordinates.

You set the Coordinate System button to Global or Local to determine relative to which system rotation will occur. Press the To Coordinates button. The Move to Coordinates dialog box appears displaying the status Local or Global.




Enter the position or rotation specification or both and press the Apply button. If all three values in a position or orientation specification are present, the specification is considered absolute; otherwise it is considered incremental. An absolute rotation specification produces three separate rotations around the current global or local axes in the sequence around X, around Y, and last around Z. You may find rotation easier to control with incremental specifications.

The Clear button removes specifications and provides an opportunity for changing between absolute and incremental modes.

## Restoring Positions After Translation or Rotation

To restore the initial state of objects or axes after motion, press the following buttons in the Translate/Rotate Selection List dialog box.

- Reset Position reverses translations.
- Reset Orientation reverses rotations.

The effects of the Reset Position button and the Reset Orientation button depend on the state of the Modify Axes Only and Use Local CPL buttons.

- Modify Axes Only inactive — Local axes of all objects in the selection list are rotated to match the global axes. The same motion applies to the axes of all objects in the list.
- Modify Axes Only active — Local axes of all objects in the selection list are rotated to match the global axes, but the objects do not move.
- Use Local CPL inactive — Local axes receive their new orientations from the global axes.
- Use Local CPL active — The current orientation of the selection is applied to the local axes.

# Path Editor

The Path Editor enables you to generate paths and Gpaths within a 3D scene. Sectioning planes, objects, and the camera can traverse paths; sectioning planes and objects can traverse Gpaths. You can define any number of paths or Gpaths in a EPD.Visualizer session. Each path or Gpath must have a unique name and an assigned color.

A path is defined by a collection of points defined by three floating point values that represent an XYZ location. The points are considered to exist in a sequence, and following that sequence describes a series of line segments. The meeting of two such line segments is called a *vertex*. An object or section plane can move along such a path. In an ordinary path, called a *path* in this document, the orientation of the object depends on the shape of the path. It is possible to define the orientation of the object relative to the angle between the two consecutive path segments at each vertex.

A Gpath allows better simulation of the manufacturing environment than a path, because in a Gpath you can describe the orientation of the object or plane at each vertex relative to a global referential rather than the characteristics of vertices. A Gpath with a single vertex is valid, but a path must have at least two vertices. The major use for a Gpath with a single vertex is giving an object a precise position and orientation.

A Gpath has the following properties:

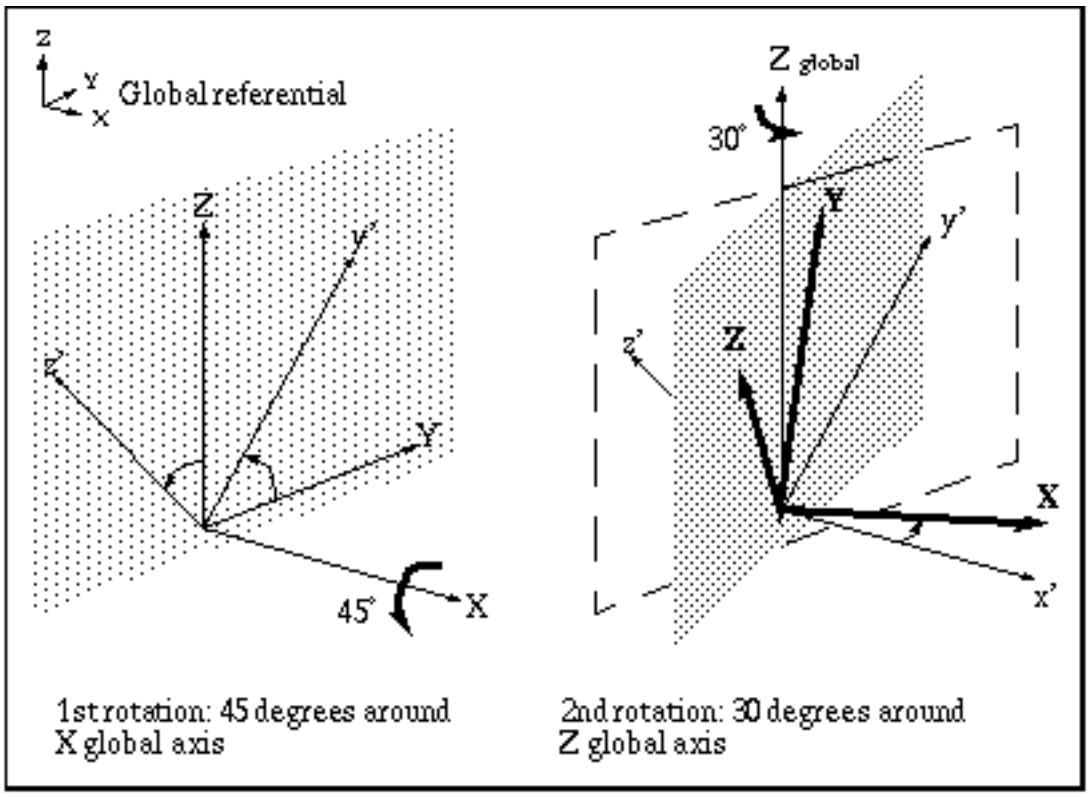
- It can be defined a in gaf file.
- It has a maximum of 500 defining points, each of which is called a *reference*.
- Each defining point has a name.
- Each defining point has an attached orientation or referential.

You define and modify Gpaths with the Path Editor. You can hide or display their references. You can modify the size of displayed references in the Path Editor window.

Each reference in a Gpath has the following defining properties:

- Its location in the 3D space
- Three rotation angles that describe how to obtain it by rotating the global referential around its X, Y, and Z axes
- Name
- You cannot operate on a reference independent of the Gpath that includes it. Some functions can use a reference to orient an object.

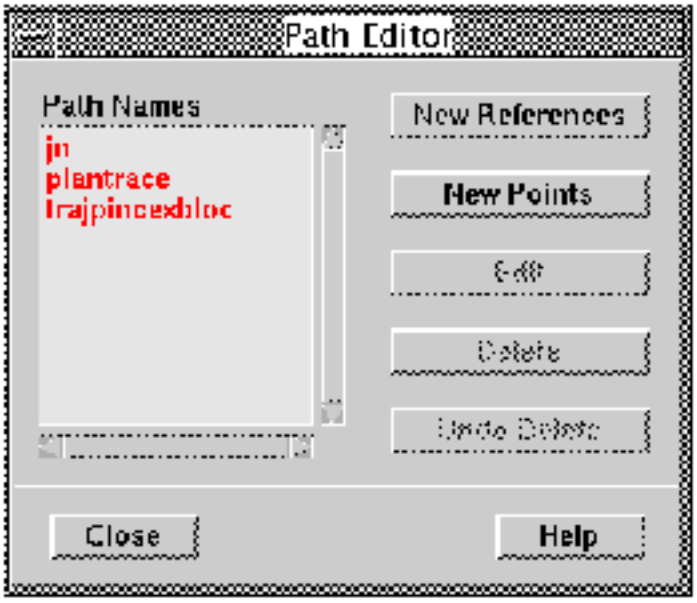
For example, a referential defined by the three angles “45, 0, 30” is obtained by successively rotating the global referential around X (45 degrees) and Z (30 degrees) as the following figure demonstrates.



## The Path Editor

To delete a path or Gpath or to begin creating or editing a path or Gpath, make selections in the Path Editor dialog box.

Select Edit > Path Editor. The Path Editor dialog box appears.

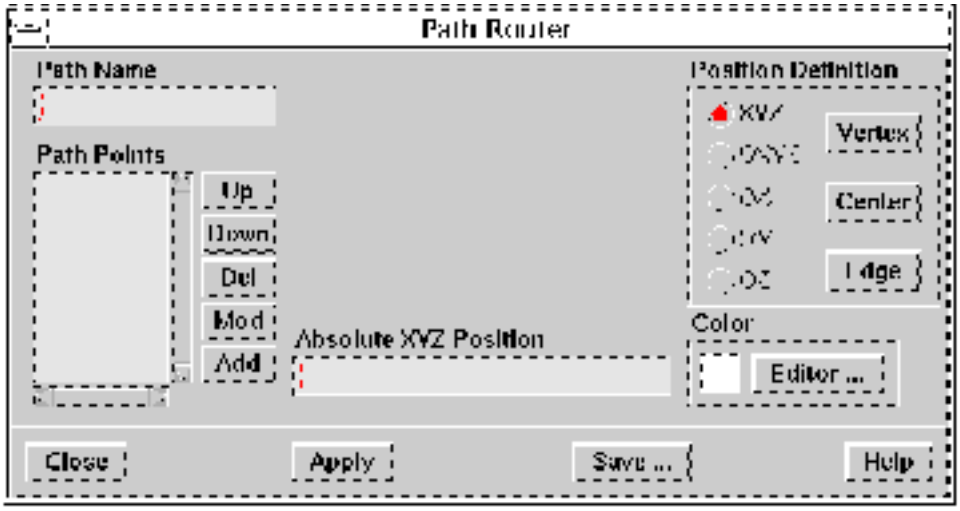


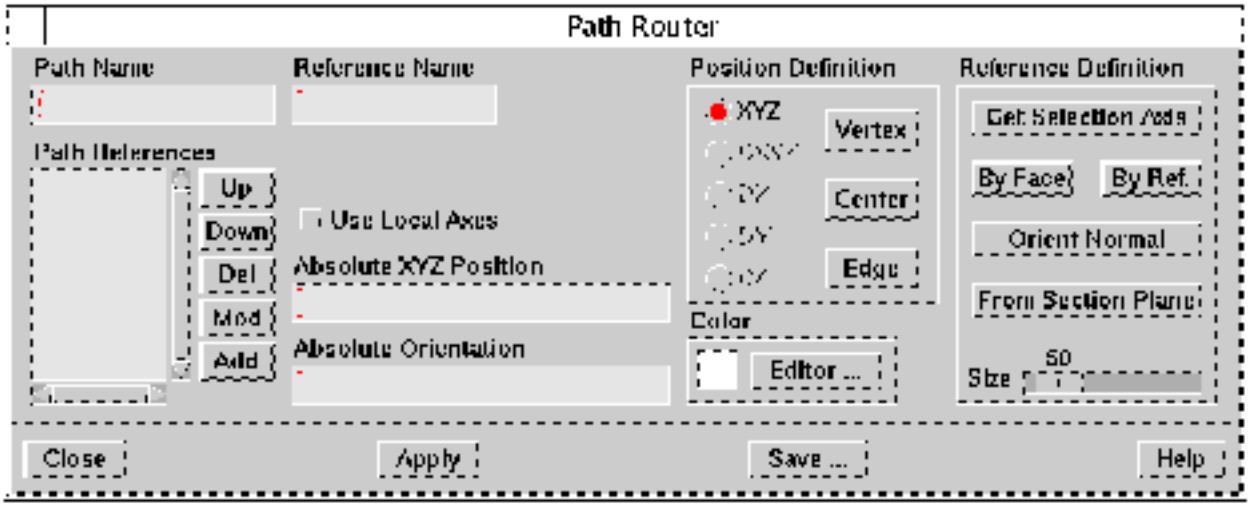

The Path Editor dialog box presents the following selections:

- New References — Click New References to create a Gpath. The Path Router dialog box displays the more complex of its two forms, which is discussed in the next section.
- New Points — Click New Points to create a path. The Path Router dialog box displays the simpler of its two forms, which is discussed in the next section.
- Edit — Select a path or Gpath in the scrolling list and click Edit to edit the selected path or Gpath. The appropriate form of the Path Router dialog box appears. The next section discusses the Path Router dialog box.
- Delete — Select a path or Gpath in the scrolling list and click Delete to delete the selected path or Gpath.
- Undo Delete — If you have deleted a path or Gpath, clicking Undo Delete reinstates the path or Gpath. You cannot restore the deleted path after you define a new path, or edit a path.

## The Path Router

The Path Router dialog box allows you to edit or create paths or Gpaths. The Path router appears in two forms: a complex form for editing or creating Gpaths and a simple form for editing or creating paths. Because the complex form includes the features of the simple form, this section discusses both forms by describing the complex form and noting where the two forms diverge.




1. Enter a path name (up to 15 characters) in the Path Name field. If you are editing, you can change the name of the path or Gpath to work on a copy.
2. Set path color with the Color Editor. Refer to “Color Editor” on page 1-19.
3. Create a path point or reference either by digitizing or by entering the point position or offset directly in the Absolute XYZ Position field. Path points appear in the Path Points scroll box in the path form, and path references appear in the Path References scroll box in the Gpath form.

Select from the following tools to control how you define the Absolute XYZ Position field locations of path points or path references. The tools also control the effect of the 3D locations that you select in the scene; the DX, DY, and DZ incremental modes allow only the X, Y, or Z components of vectors to serve as specifications between path points or references.

- Position Definition Projection modes:

  — XYZ: Absolute xyz location.

  — DXYZ: Relative distance from the currently selected point along the XYZ axis. For a selection made in the 3D scene this mode is identical to XYZ mode.

  — DX: Relative distance from the currently selected point along the X axis. When you select this mode, the new point changes only in the X value. Y and Z values remain constant.

  — DY: Relative distance from the currently selected point along the Y axis. When you select this mode, the new point changes only in the Y value. X and Z values remain constant.

  — DZ: Relative distance from the currently selected point along the Z axis. When you select this mode, the new point changes only in the Z value. X and Y values remain constant.

- Position Definition Digitizing tools:

  — Vertex: Snap to the end vertex of the edge you digitized.

  — Center: Snap to the center of the edge you digitized.

  — Edge: Snap to exactly where you digitized.

In creating Gpaths, you make a list of path references, each of which includes information in the Absolute XYZ Position field and information in the Absolute Orientation field. Select one of the following tools in the Reference Definition area to automatically create the information for references. This information controls the orientation and position of the traversing object:

- Get Selection Axis — When you translate or rotate an object belonging to the selection list, this tool makes the object's position and orientation into specifications for the references that you create.
- By Face — The reference is derived from the position and orientation of the face of an object selected in the graphics window.
- By Ref. — The position and orientation are copied from the reference that you select in the graphics window.
- Orient Normal —The orientation of the reference is derived from the perpendicular from the face of an object selected in the graphics window. The position is the current position.
- From Section Plane — The reference is copied from the local referential of the current sectioning plane.

When an incremental mode (DXYZ, DX, DY, or DZ) is selected you can specify rotation around the selected axes by making entries in the Absolute Orientation Field. The rotation occurs around local axes if you select Use Local Axes. If Use Local Axes is not selected rotation occurs around global axes. The Size selection controls the size of the reference as it is represented in the graphics window.

4. Edit the path points or path references.

   All path points or path references are displayed in the scrolling list. Next to the list are three buttons: Up, Down, and Del. The buttons allow you to reorder the list items by moving them up or down, or deleting them.

   The Mod and Add buttons allow you to use position and orientation information that you type in the Absolute XYZ Position and Absolute Orientation fields to create a new reference or to modify a selected reference. When an incremental mode (DXYZ, DX, DY, or DZ) is selected these fields are empty.

   - Absolute XYZ Position (available in both path and Gpath forms)
   - Absolute Orientation (available in Gpath form only)

   a. You select from the scrolling list a path point or path reference on which to employ the information.

   b. You select the information by selecting a path point or path reference in the graphics Window.

   c. If you do not name a path reference in the Reference Name field, its default name has two parts:

      — The value of the PVS_Unnamed_Ref environment variable is the name's root.

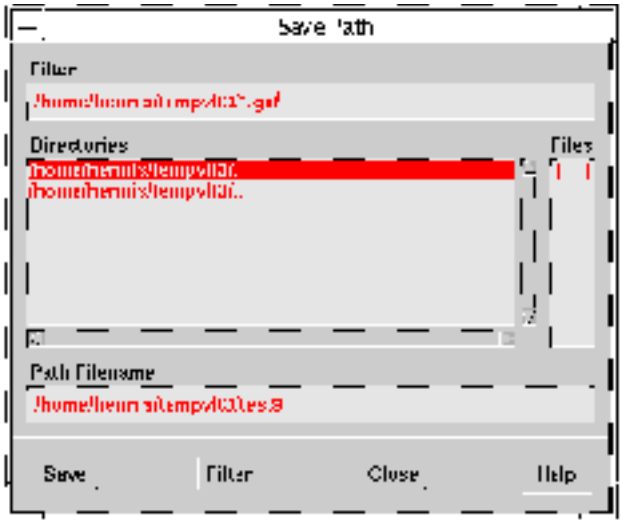
      — The line number of the path reference in the list box is the name's suffix.

   d. Clicking Mod applies the information to the path point or path reference in the scrolling list. Clicking Add places a new path point or path reference after the one selected in the scrolling list. The new path point or path reference has the selected information.

5. Complete your work in the Path Router:

- Close — Closes the window without applying or saving work.
- Apply — Applies the path or Gpath data that does not appear in the graphics window but exists in the Path Router.

- Save — Displays the Save Path dialog box and also performs the operation that the Apply button performs.




The Save Path dialog box allows you save the path or Gpath to a `.gaf` file. The Save Path window offers the functions for saving described in "File Selection Dialog Box" on page 1-17, except that the Save Path window has a Path Filename field rather than a Selection field. You type the location for saving the path or Gpath in the Path Filename field. If you do not type the suffix `.gaf`, it is added automatically. If you specify non-existent directories, they will be created if permissions permit.

The environment variable PVS_PATHS_DIR initializes the Filters field and the Directories list. The environment variable PVS_PATHS_DIR is read only when EPD.Visualizer starts or when you select File > New Session in the EPD.Visualizer window.

# Visual Properties

Use visual properties to define the following:

- Item Color — Change the color of items in the 3D scene
- Surface Materials — Assign properties to the surfaces in the 3D scene.

## Item Color

With this option, you can change the color of items in the 3D scene that are on the selection list.

To change item colors in the 3D scene, use the following steps:

1. Pick up the Select tool from the palette and select the item(s).
2. Select Edit > Visual Properties > Item Color and the Item Color Editor dialog box displays.

3. Refer to "Color Editor" on page 1-19 to use the Editor button.
4. Click Apply to change the item color to the new color.
5. Click Close when you finish changing colors of other selected items in the 3D scene.

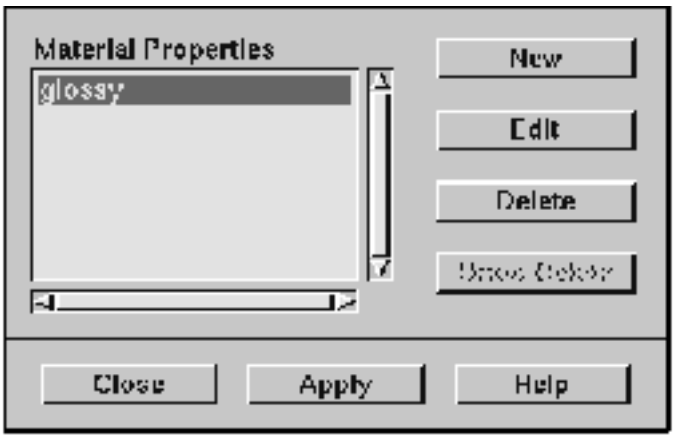
## Surface Materials

With this option, you can assign properties to the surfaces in your 3D scene. Material properties can include the following:

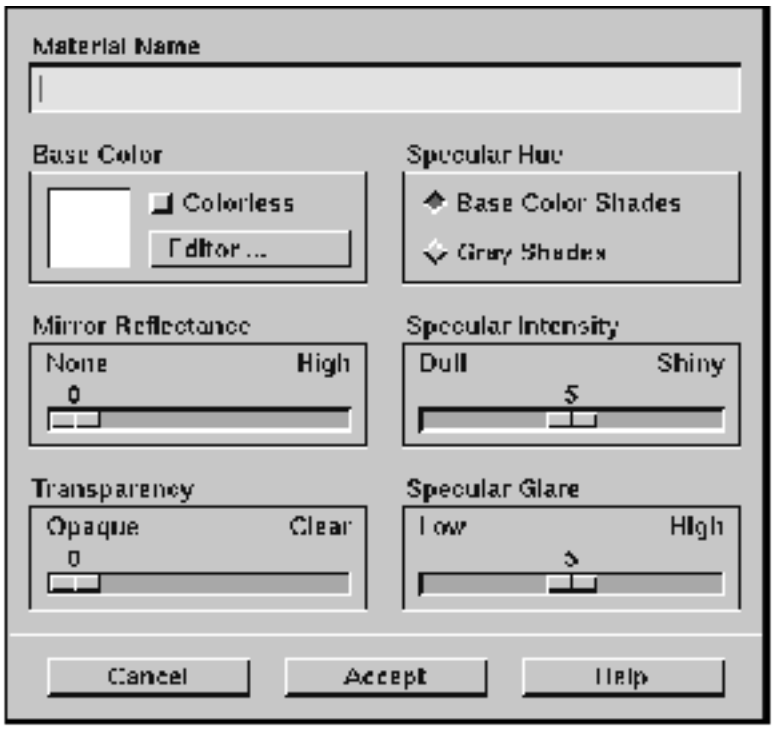
- Base Color — The color or non-color of a surface or material.
- Mirror Reflectance — The reflective nature of a material. Wood, for example, is non-reflective, whereas chrome is highly reflective.
- Transparency — The degree of transparency of a 2D surface.
- Specular Hue — The color of the reflection of the material. Metallic materials typically have colored specular reflections, while non-metallic materials typically have specular reflections in shades of black and white.
- Specular Intensity — The intensity of a reflection. Dull surfaces appear matted or flat, due to the scattering effect of the light after hitting a surface. Shiny surfaces appear hard and polished due to the strong directional component of reflected light.
- Specular Glare — The focus of specular highlights off a surface from light sources. A material with high specular glare produces a small focused shiny spot on curved surfaces; a material with low specular glare produces a shiny spot that is broader and more scattered.

To create a new material property, use the following steps:

1. Select Edit > Visual Properties > Surface Materials. The following figure shows the Material Selection dialog box.




2. Click New and the Material Editor dialog box displays as shown in the following figure.




3. Set the following options as necessary:
   - Material Name — Enter a name for the material in the Material Name box.

- Base Color — Set the base color of the material, or click the Colorless button to maintain the original color of the object.
- Mirror Reflectance — Use the Mirror Reflectance scale to set the level of mirror reflectance in a material. Mirror reflectance is used in ray tracing operations only.

  — None: Results in a non-reflective surface

  — High: Results in a highly reflective surface
- Transparency — Use the Transparency scale to specify a transparency factor. Transparency is used in ray tracing operations only.

  — Opaque: Results in an opaque, or non-transparent, surface

  — Clear: Results in a clear or transparent surface
- Specular Hue — Set the specular hue of the material.

  — Base Color Shades: Results in colored specular reflections

  — Gray Shades: Results in specular reflections in shades of gray
- Specular Intensity — Use the Specular Intensity scale to specify whether a material is dull, shiny, or in between.

  — Dull: Results in a dull material surface

  — Shiny: Results in a shiny material surface
- Specular Glare — Use the Specular Glare scale to control the degree of focus of specular highlights (shiny spots) of a material.

  — High: Results in a material with highly focused specular highlights

  — Low: Results in a material with broad specular highlights

4. Click Accept to save your material definitions.

To assign material definition to an object, use the following steps:

1. Click a material definition from the Material Properties list in the Material Selection dialog box.
2. Pick up the Select tool from the palette and select the object(s) with planar surfaces.
3. Click Apply to assign the material to the object(s).

To select material pick up the Select tool from the palette and select an object to retrieve its material. If the object you select has material assigned to it, the definition for that material is highlighted in the Material Properties box.

To edit a material definition, use the following steps:

1. Select the material.
2. Click Edit.
3. Edit the settings as needed.
4. Click Accept to save your changes.

To delete a material, use the following steps:

1. Click the material name in the Material Properties list.
2. Click Delete and the material is removed from the Material Properties list and temporarily stored.

To restore deleted material, click Undo Delete to restore the material.

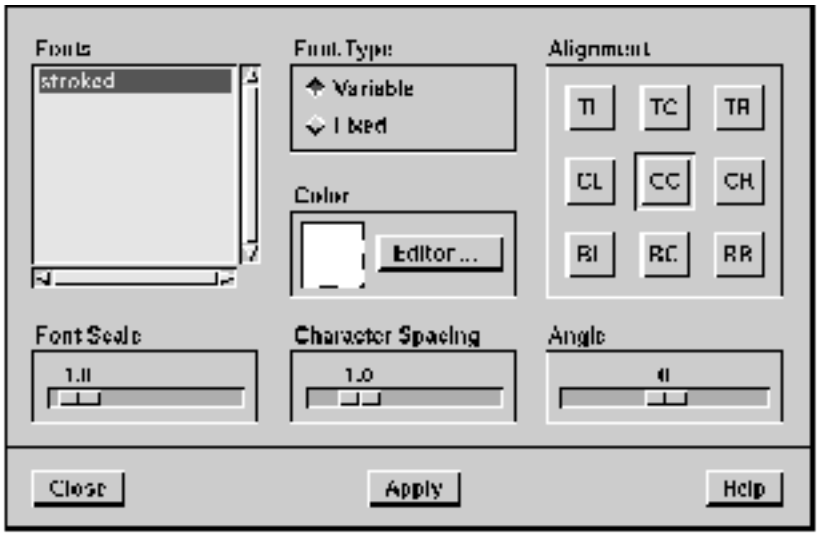
Please note: You cannot restore the deleted material after you close the dialog box, define a new material, or edit a material.

## Text Properties

With this option, you can define the font, font type (variable or fixed), alignment, font size, text color, spacing between characters, and text angle. Any newly-created text or text on the selection list adopts the properties you set in the Text Properties dialog box.

To set the text properties, use the following steps:

1. Select Edit > Text Properties. The following figure shows the Text Properties dialog box.




2. Set the following options as necessary before with the Text and Object Tagging tools:
   - Font Type

     — Variable: Define the specific font size with the Font Scale Factor slider scale

     — Fixed: Select a predefined font type and size
   - Fonts — Click the name of a font
   - Alignment — Set the alignment of the text based on the digitize marker from one of the following options:

     — TL (top-left)

     — TC (top-center)

     — TR (top-right)

     — CL (center-left)

     — CC (center-center)

     — CR (center-right)

     — BL (bottom-left)

     — BC (bottom-center)

     — BR (bottom-right)

     For example, if you select TR, the digitize marker appears at the top right of the text. The default is CC (center-center).

- Color Editor — Set the text color. To use the Color Editor, refer to the "Color Editor" on page 1-19.
- Character Spacing — Set the amount of space between each character with the Character Spacing slider scale. A spacing value of 1.0 means that the normal spacing for the selected text is used. A value of 2.0 results in two times the normal spacing. A spacing value of 0.0 causes the letters to stack on top of each other. The default value is 1.0.
- Angle (Degrees) — Set the Angle slider scale to define the direction and angle of text. The default is zero.

  Text with an angle of zero displays as follows:

```
TEXT
```

  Text with an angle of 45 degrees displays as follows:

```
      T
    E
  X
T
```

Text with an angle of -45 degrees displays as follows:

```
T
  E
    X
      T
```

3. Click Apply and Close after you set the properties.

## Delete

After you select item(s), you can delete them by selecting Edit > Delete. You are prompted with the following message:

```
All items on the selection list will be deleted!
Click 'OK' to delete...
```

To delete the item, click OK. If you do not want to delete the item, click Cancel.

# View Menu

By selecting View on the menu bar, you can access the following menu items:

- Primary View — Set and show the primary view
- View — Center and zoom the 3D scene within the limits of the coordinates of the items
- Change View — Change to an orthogonal view, a view defined by two points, or switch between orthographic and perspective views.
- Secondary Views — Create secondary views
- Perspective — Change perspective angle of the camera
- Rendering — Select rendering method of objects
- Lighting — Define, duplicate, and edit lights and control scene brightness
- View Editor — Define views and save notes
- View Sequence — Define view sequences and play them back
- Camera Traversal — Traverse the camera along defined paths
- Movie Recorder — Define movie parameters/controls

The View menu items enable you to manage the display of items in your 3D scene, define views based on camera position, and use camera traversal to move through the 3D scene.

## Primary View

The primary view is a view that you can return to if you become disoriented while moving through the 3D scene. The initial primary view is defined when you add your first graphic file. The primary view is saved with a metafile and is later restored when you open a metafile. You can also set a new primary view.

### Show

To show the primary view, select View > Primary View > Show.

### Set

To set a new primary view, define the view and select View > Primary View > Set.

# View

To display all items centered and within the limits of the graphics window, select View > All. You can also view only the visible items by selecting View > Visible.

## All

To view all items regardless of visibility, select View > All.

## Visible

To view only the visible items, select View > Visible.

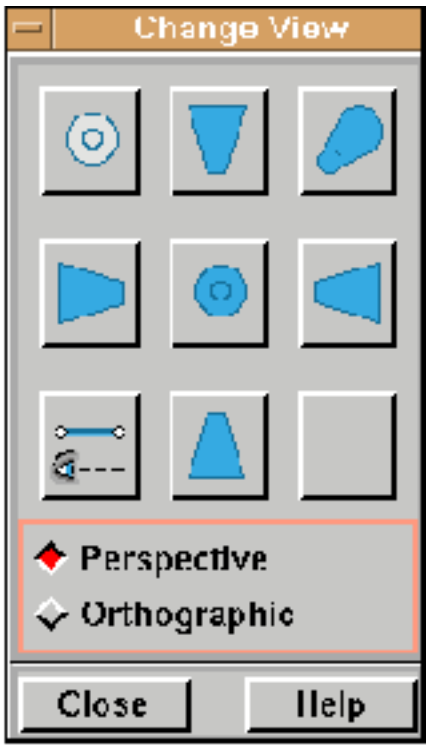
Please note: Because there is no visibility checking done for View > All, this option is much faster than View > Visible.

# Changing Views

The Change View option provides the following:

- Seven main orthogonal views, which are automatically computed when you set the camera position and target
- A view defined by two points selections
- Orthographic and perspective view representations

To change your view, select View > Change View. The Change View dialog box appears:




The views available in the Change View dialog box are defined in the following table:

| Row | Orthogonal Views | | |
|---|---|---|---|
| Top | Rear | Top | Isometric |
| Middle | Left | Front | Right |
| Bottom | 2 point | Bottom | |

The view representation mode resets in the following cases:

- When you select Perspective in the dialog box
- When you select View > Perspective to choose a perspective angle
- At EPD.Visualizer startup or when selecting File > New Session begins a new session, the view representation mode resets according to the PVS_ORTHO_MODE environment variable:
  - If the PVS_ORTHO_MODE environment variable is not set, or is set to zero, the Perspective mode is selected.
  - If the PVS_ORTHO_MODE environment variable is set to a non-zero value, the Orthographic mode is selected.

Click the Close button to cancel the operation and menu.

## Secondary Views

You can create two secondary views as smaller versions of the main view. Secondary views are useful for viewing dynamics. Entity selection is supported only in the main view.

To create a secondary view, select View > Secondary Views > Create. By default, EPD.Visualizer scales the secondary views at one quarter width of the main view. To customize the default secondary view scale, set the PVS_SECOND_VIEW_SIZE environment variable in the range from 0.1 to 0.5. The scale factor is relative to the width of the main view in the graphics Window. Refer to “PVS_SECOND_VIEW_SIZE” on page H-3 for further information about this environment variable.

To manipulate the upper of the two secondary views, select View > Secondary Views > Upper View, then proceed as you would for the main view. Select Lower View to manipulate the lower view, and Main View to return to manipulating the main view.

You can display an orientation box around the secondary views to visually orient the views to their position relative to the main view. To display a 2D box or 3D pyramid for your secondary views, select View > Secondary Views > Position Display > 2D / 3D.

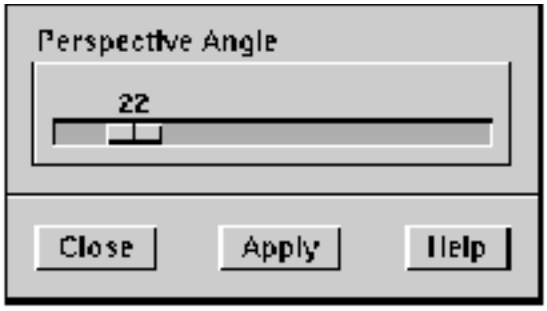
By default, the color of the orientation box is black or white, depending on your background color. You can switch between this default and a color of your choice by selecting Options > Temp Color > User. This command specifies to use the

color defined by the PVS_TEMP_COLOR environment variable. Refer to “Temp Color” on page 3-75 for further information on setting this variable.

To delete secondary views, select View > Secondary Views > Delete. Note that EPD.Visualizer removes secondary views when performing radiosity or ray trace display.

## Perspective

To increase or decrease your viewing area (similar to a wide angle lens), select View > Perspective. The Perspective Definition dialog box displays as shown in the following figure.



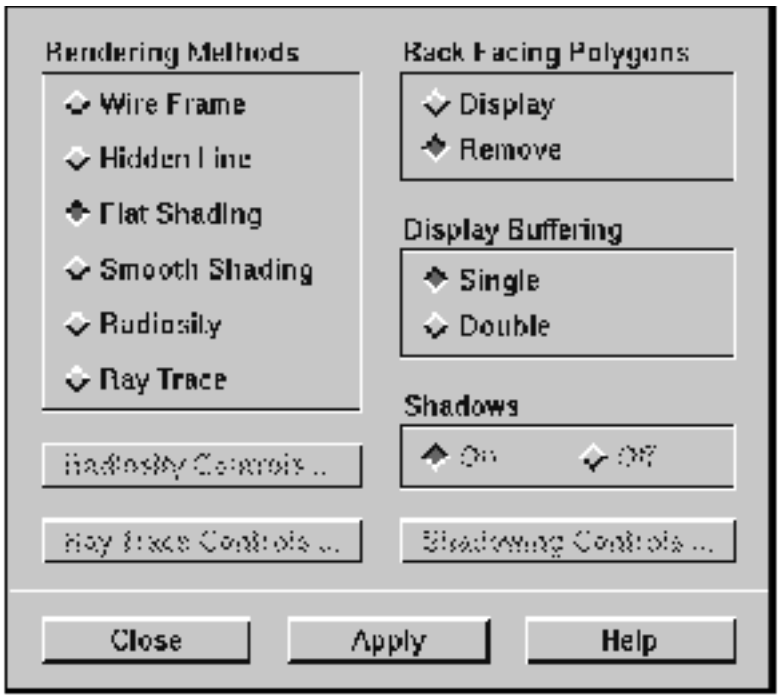

Use the slider scale to set the perspective angle to increase or decrease your viewing area.

## Rendering

EPD.Visualizer enables you to generate realistic scenes with a combination of material properties, lighting, and rendering methods. For a general discussion of rendering methods, refer to “Rendering” on page 2-2.

To render an image, use the following steps:

1. Select View > Rendering. The following figure shows the Rendering Control dialog box.




2. Set the following options as necessary.

Please note: Since it can be time consuming to render the screen after each rendering change, you can set as many rendering controls as you need, then apply them all at once.

Rendering Methods

- Wire Frame — Display all surface edges of the 3D scene as lines. Wire Frame displays all lines in the 3D scene, regardless of the visibility of the lines.
- Hidden Line — Display a solid view of the 3D scene with black polygon edges and lights off.
- Flat Shading — Display a shaded solid view of the 3D scene. The edges of the facets between the polygons are distinguishable.
- Smooth Shading — Display a smooth shaded solid view of the 3D scene. Polygons that are defined as shells in the graphic files display so that you do not see the edges between them.
- Radiosity — Display the light that is reflected or radiated between diffusely reflecting surfaces in a 3D scene. Radiosity produces view

independent images. You can change the view angle without performing the radiosity operation again.

If you select Radiosity as the rendering method, to set the appropriate options click the Radiosity Controls button. (Refer to "Radiosity Controls" on page 3-50 for more information.)

Please note: Radiosity does not handle specular lighting and cannot display specular highlights, reflections, or refraction.

- Ray Trace — Display reflections and refractions in the 3D scene. Ray tracing shows reflections, mirroring, transparency, and refraction. Ray tracing produces transient, pixel based images. Therefore, you might want to keep window sizes small to reduce the amount of time it takes to generate the ray traced image.

  If you select Ray Trace as the rendering method, to set the appropriate options click the Ray Trace Controls button. (Refer to "Ray Trace Controls" on page 3-48 for more information.)

Please note: Ray tracing does not handle matte or diffusely reflecting surfaces.

Back Facing Polygons

- Display — Polygons facing the back that you cannot normally see are included in the display.
- Remove — Polygons facing the back that you cannot normally see are removed from the display.

Please note: Removing back facing polygons works well when the objects are completely closed. However, if objects contain holes or openings, the back facing polygons that you ordinarily see through the openings are invisible.

Display Buffering

- Single — Each item displays in the graphics window as it is rendered.
- Double — Items do not display in the graphics window until they are all completely rendered.

Please note: For a more realistic view, use double buffering when performing such operations as camera or item traversal and any walk-through operations.

Shadows — With radiosity or ray tracing, you can control the display of shadows.

- On — Displays shadows in the 3D scene.
- Off — Makes shadows invisible. With Shadows set to off, the 3D scene is illuminated by light only.

To specify which objects cast or receive shadows, select the object(s) and click the Shadowing Controls button. (Refer to “Shadowing Controls” on page 3-48 for more information.)

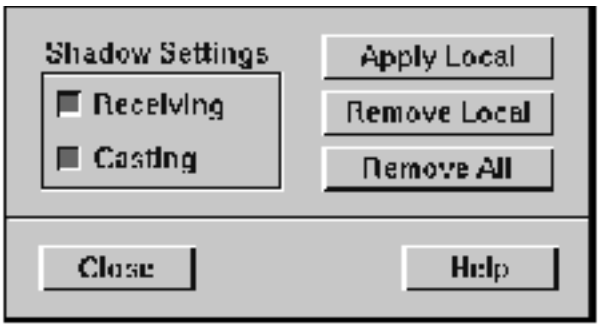
3. Click Apply to see the effects of the rendering on your 3D scene.

## Shadowing Controls

If you select radiosity or ray trace as the rendering method, you can control shadows on an object-by-object basis.

To set shadowing controls, use the following steps:

1. Select the object(s) for which you want to inhibit the casting or receiving of shadows.
2. Click the Shadowing Controls button. The Shadowing Controls dialog box displays.




3. Set the following options:
   - Shadow Settings

     Receiving: Controls whether or not an object can receive shadows. The default is On.

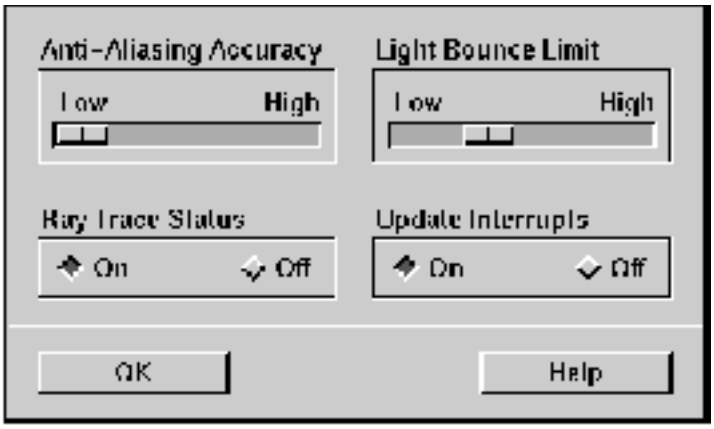
     Casting: Controls whether or not an object can cast shadows. The default is On.
   - Apply Local — Apply shadow controls to object(s) on the selection list.
   - Remove Local — Remove shadow controls from object(s) on the selection list.
   - Remove All — Remove shadow controls from all object(s).

## Ray Trace Controls

If you select ray trace as the rendering method, you can control the quality of the image and the rendering speed during ray trace calculations.

To set ray tracing controls, use the following steps:

1. Click Ray Trace Controls in the Rendering Control dialog box. The following figure shows the Ray Tracing Control dialog box.




2. Set the following options:
   - Anti-Aliasing Accuracy — Use the Anti-Aliasing Accuracy slider scale to set the level of anti-aliasing. Anti-aliasing controls the stair stepping effect of surface edges. The lower the anti-alias accuracy, the greater the stair stepping effect. You can generate fairly realistic scenes with low anti-aliasing levels.

     Increasing anti-aliasing lengthens the time to perform a ray trace calculation.
   - Light Bounce Limit — Use the Light Bounce Limit slider scale to set the light bounce limit. Light bounce controls the number of times a ray of light can bounce around a 3D scene. You can generate fairly realistic scenes with a low light bounce limit.

     Increasing the light bounce limit lengthens the time to perform a ray trace calculation.
   - Ray Trace Status — Turn ray trace status on or off. When ray trace status is on, a graduated slider scale displays that shows progress during a ray trace calculation. The default is On.
   - Update Interrupts

     — On: Enables you to interrupt the ray trace rendering operation.

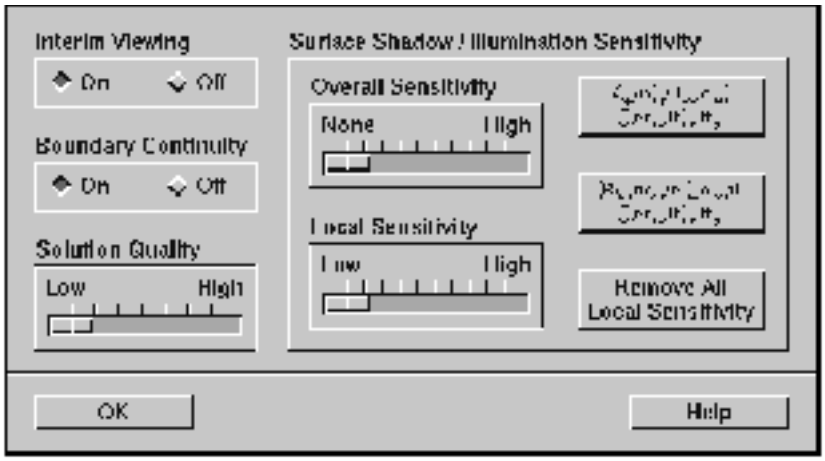
     — Off: Improves system performance, but you must wait until the rendering process is complete before you regain control of the user interface.
3. Click OK to accept ray tracing control settings.

## Radiosity Controls

If you select radiosity as the rendering method, you can control the quality and realism of the image.

To set radiosity controls, use the following steps:

1. Click Radiosity Controls in the Rendering Control dialog box, and the Radiosity Control dialog box displays.




2. Set the following options as necessary:
   - Interim Viewing

     — On: Renders the screen at each periodic radiosity solution step. This enables you to terminate additional calculations and keep the current display.

     — Off: Renders the screen either when the radiosity calculation is complete or when you click Stop on the radiosity status popup menu.
   - Boundary Continuity (On) — Forces continuous shading across common edges of surface facets. The radiosity calculation goes more slowly when you turn on boundary continuity.
   - Solution Quality — Sets the amount of reflected light that EPD.Visualizer accounts for in the radiosity solution.

     — High: Results in the highest quality radiosity solution; it accounts for all direct and radiant light energy. A high solution quality setting significantly slows down radiosity calculation time.

     — Low: Results in lower quality radiosity solutions. You can achieve excellent results by setting solution quality between low and medium, since the radiosity calculation considers the most important light energy contributors first.

- Overall Sensitivity — Controls the shadow and illumination of all surfaces without their own local sensitivity settings.

  — None: Results in no global sensitivity.

  — High: Results in the maximum amount of sensitivity to shadows and light.

- Local Sensitivity — Controls the shadow and illumination sensitivity of items on the selection list. After setting local sensitivity, apply your changes with the local sensitivity buttons described below.

  — Apply Local Sensitivity: Adds local sensitivity of items on the selection list. A warning message displays if the selection list is empty.

  — Remove Local Sensitivity: Removes local sensitivity from items on the selection list. A warning message displays if the selection list is empty.

  — Remove All Local Sensitivity: Removes local sensitivity from all items in a 3D scene.

3. Click OK to accept radiosity control settings.

# Lighting

The lighting option enables you to manage, define, and modify the lighting for your 3D scene.

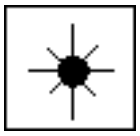
## Editor

You can define and manage multiple directional, point, and spot lights with the Light Editor. You can modify the light settings, turn lights on and off, duplicate lights, either singly or in a matrix, and show or hide light geometry.

Directional Lights: A directional light is infinitely far away from the objects it illuminates. All light rays from a directional light are parallel and uniformly illuminate all surfaces in your 3D scene.

Point Lights: A point light is like a tiny light bulb. It occupies one point and radiates light equally in all directions. Figure 3-1 shows the symbol for a point light:

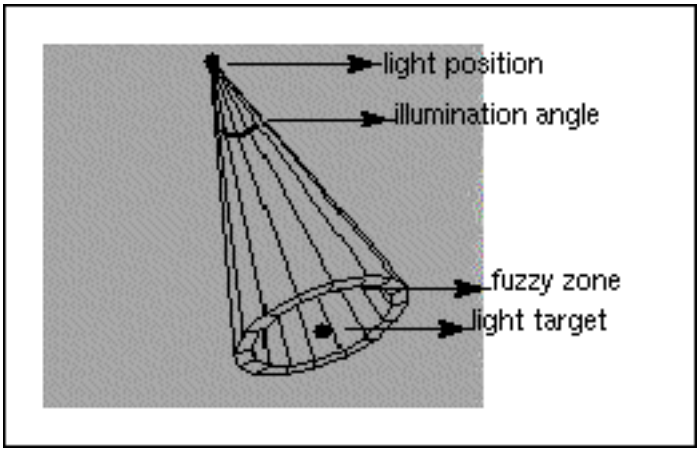
**Figure 3-1 Point Light Symbol**



Spot Lights : A spot light is a type of point light. You can specify a sharp-edged spot light, a variable-focus flashlight, or a fuzzy-edged flood light.

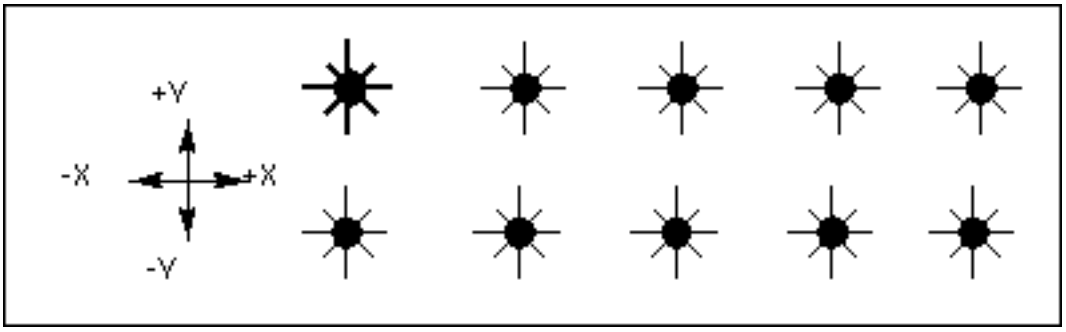
When a spot light displays, the vertex of the cone is at the light position, and the center of the circle is at the light target. The area between the inner circle and the outer circle shows the fuzzy zone of the spot light. The following figure shows the symbol for a spot light:

**Figure 3-2 Spot Light Symbol**




Matrix Duplication: The Matrix option allows you to duplicate a light into a matrix of rows and columns. You can do matrix duplication of point and spot lights, but not directional lights. Figure 3-3 shows a two row, five column light matrix. The original light is shown in bold.

**Figure 3-3 Matrix Duplication of Lights**

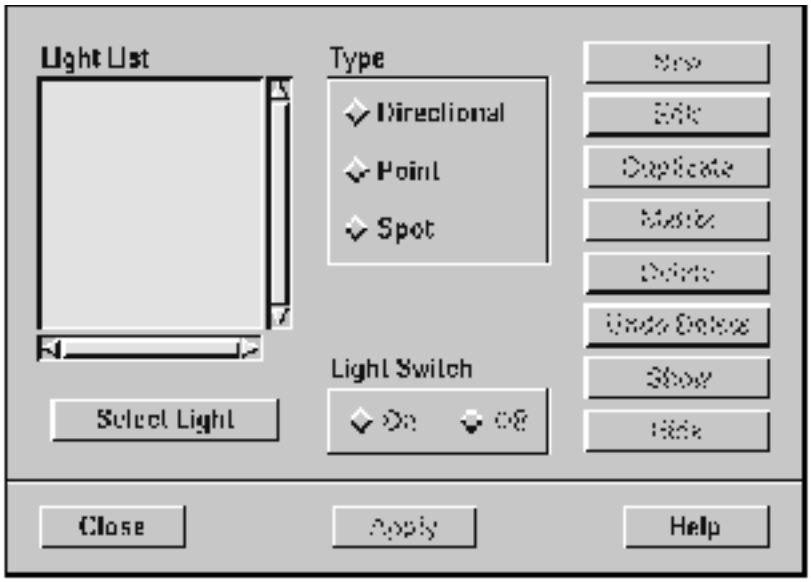



Show or Hide Light Geometry: This option allows you to show or hide the geometry of point or spot lights. You cannot show the geometry of directional lights because they are infinitely far away. Hide causes the geometry of the light to disappear from view. Show displays the geometry of the light.

The following sections describe how to define a new light; edit, delete, show/hide or duplicate lights; and perform matrix duplication.

To define a new light, use the following steps:

1. Select View > Lighting > Editor. The Light Editor dialog box displays.

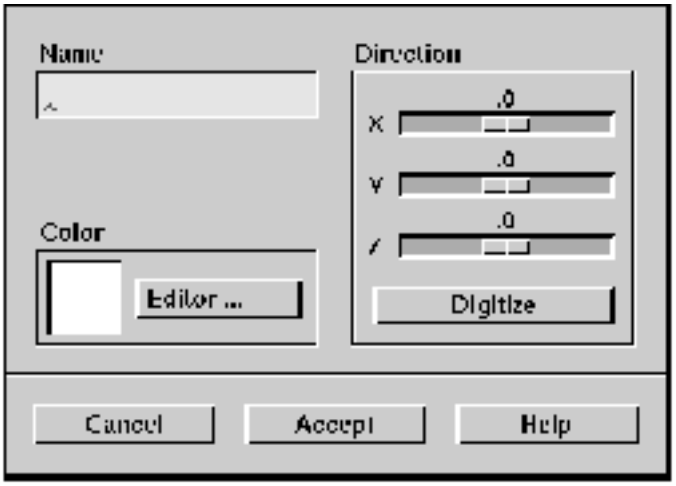



2. Set the following options as necessary:
   - Light Switch — Turn lights on and off. The default is On.
   - Type — Select the type of light: Directional, Point, or Spot lights.

3. Select a light type.
4. Click New; its editor dialog box displays.
5. Enter a name for the light in the text box.
6. Set the light color. To use the Light Color Editor, refer to “Color Editor” on page 1-19.
7. Set the option(s) for the new light as described in the following sections on defining directional lights, point lights, or spot lights.
8. Click Show for point or spot lights to view the position of the light relative to the 3D scene.
9. Click Accept to insert the light. The Light Editor dialog box redisplays.
10. Define the light
11. Click Apply to see the effects of the light on the 3D scene. You can define multiple lights, then apply them all at once.

To define a directional light, use the following steps:

1. Click Directional, then click New. The Directional Light Editor displays.




2. Name the light and set the light color.
3. Set the following options as necessary:
   - Direction — Set the direction of the light. Direction is defined by a simple vector passing through the origin of the 3D scene from the source position that you define.

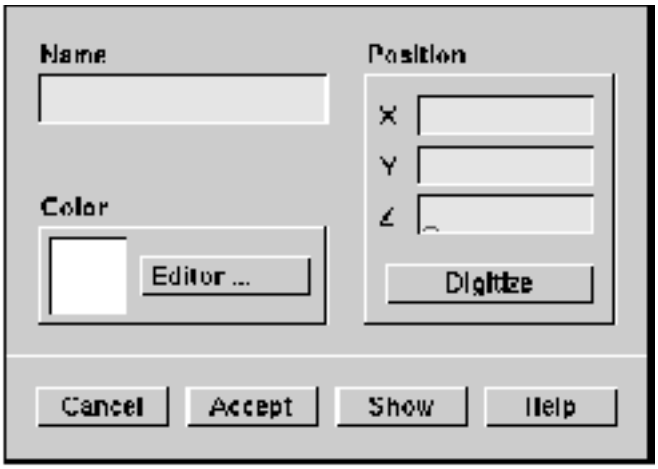
To set the light direction, you can either:

— Use the three Direction slider scales to set the xyz values. The scales range from -1.0 to 1.0.

— Use the Digitize button to digitize two model coordinates. The xyz values corresponding to the digitized vector appear in the Direction xyz slider scales.

To define a point light, use the following steps:

1. Click Point, then click New. The Point Light Editor displays.



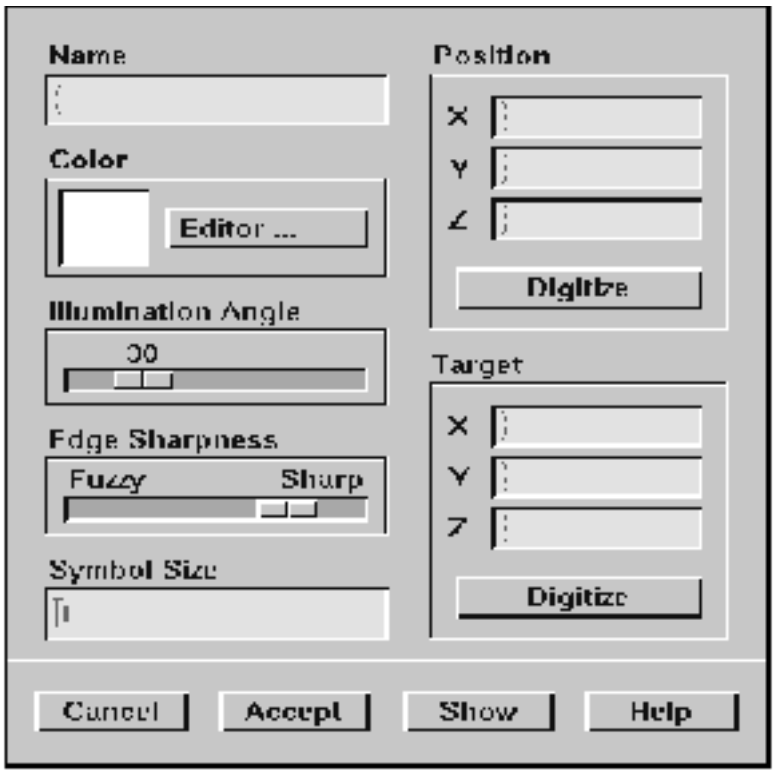

2. Name the light and set the light color
3. Set the following options as necessary:
   - Position — Set the light position. To set the light position, you can either

     — Use the three text boxes to set the xyz values.

     — Use the Digitize button to digitize a model coordinate. The xyz values corresponding to the digitize appear in the Position text boxes.

To define a spot light, use the following steps:

1. Click Spot, then click New. The Spot Light Editor displays.




2. Name the light and set the light color
3. Set the following options as necessary:
   - Position — Set the light position. To set the light position, you can either:

     — Use the three text boxes to set the xyz values.

     — Use the Digitize button to digitize a model coordinate. The xyz values corresponding to the digitize appear in the Position text boxes.
   - Target — Set the light target. To set the light target, you can either:

     — Use the three text boxes to set the xyz values.

     — Use the Digitize button to digitize a model coordinate. The xyz values corresponding to the digitize appear in the Target text boxes.
   - Illumination Angle — Use the slider scale to set the illumination angle. The illumination angle specifies the conic area that you want the spot light to illuminate. The cone size is measured from one edge of the cone, through the light-direction vector, to the opposite cone edge. The angle must be larger than 0 and less than or equal to 135 degrees. The default is 30 degrees.

- Edge Sharpness — An edge can be sharp, fuzzy, or somewhere in between. A sharp edge gives an abrupt change in brightness at the edge of the area illuminated. A fuzzy edge gives a gradual reduction of the level of brightness, so the edge appears softened. The default edge sharpness is 80 percent.
- Symbol Size — Set the size in working units of the spot light symbol. The larger the number entered, the larger the symbol displays.

To edit a light, use the following steps:

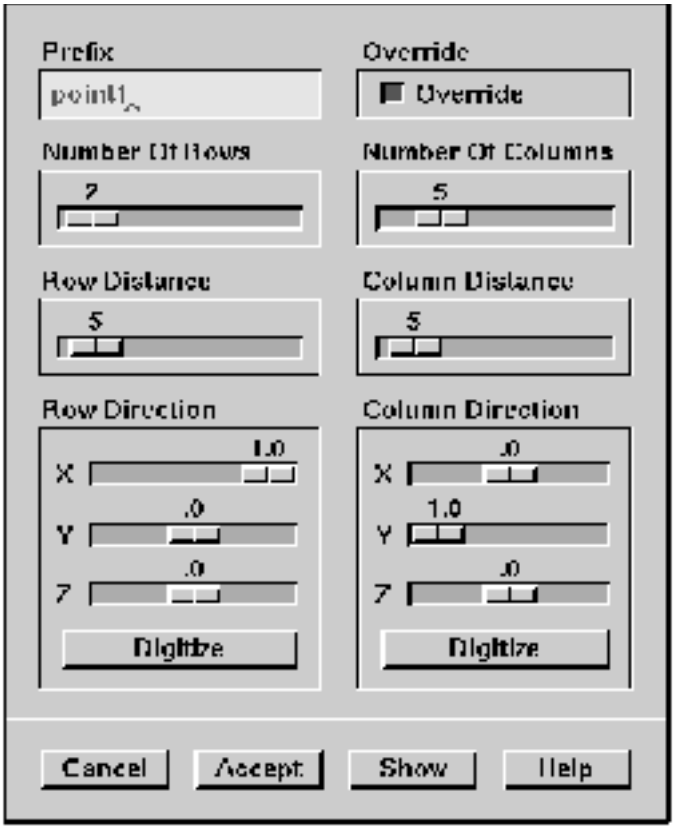
1. Click a light type in the Light Editor dialog box.
2. Click the light name you want to edit, then click Edit. The current settings for that light display in the appropriate dialog box.
3. Reset the options as necessary.
4. Click Accept to save your changes.

To duplicate a light, use the following steps:

1. Click a light type in the Light Editor dialog box.
2. Click the light name you want to duplicate, then click Duplicate. The current settings for that light display in the appropriate dialog box.
3. Enter a new, unique name in the text box.
4. Reset the options as necessary.
5. Click Accept to insert the duplicated light.

To duplicate a light in a matrix, use the following steps:

1. Click a light type in the Light Editor dialog box.
2. Click the light name you want to duplicate into a matrix, then click Matrix. The following figure shows the Light Matrix dialog box.




3. Set the following options as necessary:
   - Prefix — By default, the Prefix box contains the name of the original light. Subsequent light names are composed of the prefix plus row position and column position. The syntax for new light names follows:

     prefix_row_number_column_number

     For example, the light in the figure above has the name point1. The duplicate light created at row 1, column 2 has the name point1_1_2.
   - Override — It is possible to create lights in a matrix that have the same names as existing light names. If Override is off, you receive a warning if a new light has the same name as an existing light. The default is On.

     If Override is on, the new lights override existing lights without warning you.
   - Number of Rows — Sets the number of rows in your matrix.

- Row Distance — Sets the distance between lights in the same row.
- Row Direction — Specifies where, relative to the original light position, to place the duplicated row of lights. To set row direction, you can either:

  — Use the three Row Direction slider scales to set the xyz values. The scales range from -1.0 to 1.0.

  — Use the Digitize button to digitize two coordinates in the 3D scene. The xyz values corresponding to the digitize appear in the Row Direction xyz slider scales.
- Number of Columns — Sets the number of columns in your matrix.
- Column Distance — Sets the distance between lights in the same column.
- Column Direction — Specifies where, relative to the original light position, to place the duplicated column of lights. To set column direction, you can either:

  — Use the three Column Direction slider scales to set the xyz values. The scales range from -1.0 to 1.0.

  — Use the Digitize button to digitize two coordinates in the 3D scene. The xyz values corresponding to the digitize appear in the Column Direction xyz slider scales.

To delete a light, use the following steps:

1. Click the light name on the Light List.
2. Click Delete and the light is removed from the Light List and temporarily stored.

To restore a deleted light, click Undo Delete.

Please note: You cannot restore the deleted light after you close the dialog box, define a new light, or edit a light name.

To show or hide light geometry, use the following steps:

1. Click a light type in the Light Editor dialog box.
2. Click the light name you want to show or hide, then click Show/Hide. The geometry of that light appears or disappears.

### Environment

With the lighting environment option, you can turn the ambient light on or off and control the brightness of all lights by adjusting the Environment Brightness.

After you select View > Lighting > Environment, the Global Light Control dialog box displays.

To turn the ambient light on and off click the On or Off button. Then click Apply to display the change.

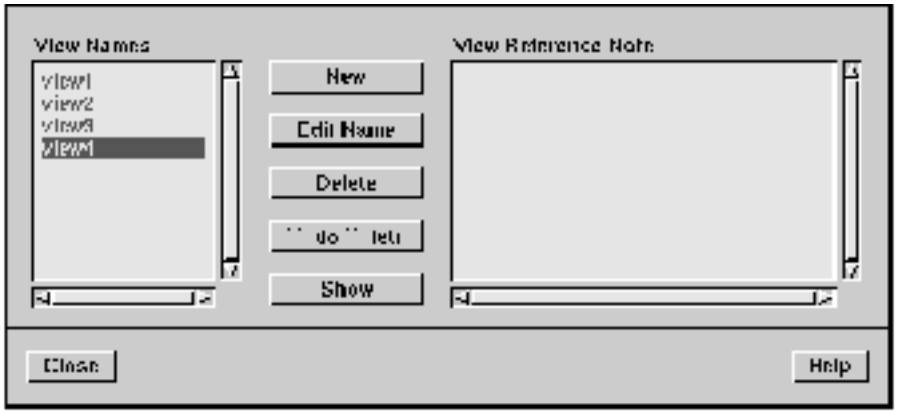
To control the brightness of the environment drag the slider on the Environment Brightness slider scale to the appropriate level. Then click Apply to display the change.

## View Editor

You can save, label, and retrieve individual views with a specific camera orientation, including camera position, camera target and perspective angle. You can save any number of views, which you can later save with a metafile.

To define a view, use the following steps:

1. Define your position in the 3D scene with the walk-through buttons.
2. Select View > View Editor. The View Definition dialog box displays.




3. Click New and a text box displays.
4. Enter a name in the text box.
5. Click OK to save the name and it displays in the View Names list.

To edit a view name, use the following steps:

1. Click the name in the View Names list.
2. Click Edit Name and a text box displays the selected name.
3. Click OK after you edit the name.

To delete a view, use the following steps:

1. Click the name in the View Names list.
2. Click Delete and the view is removed from the View Names list and temporarily stored.

To restore a deleted view, click Undo Delete to restore a deleted view. You cannot restore the deleted view after you close the dialog box, define a new view, or edit a view name.

To display a view click the view name, then click Show. The view displays in the graphics window and the note displays in the View Reference Note text box.

To enter a note, after you name a view, enter a note of any length in the View Reference Note text box.

To edit a note, select the view name and edit the text.

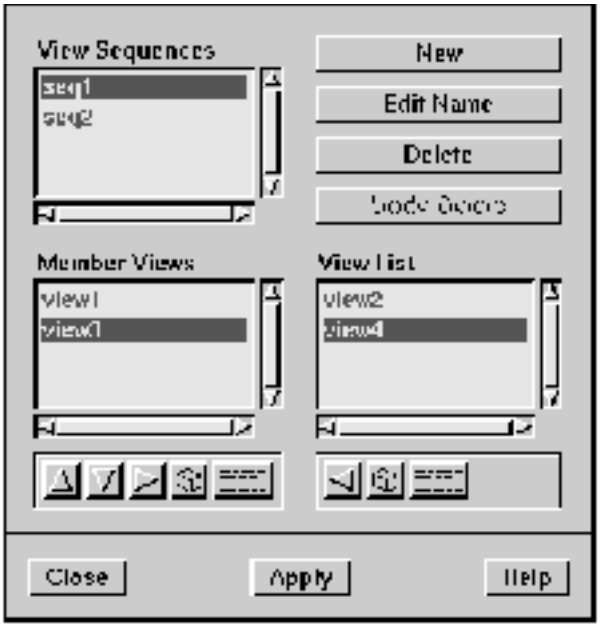
## View Sequence

After you define views, you can organize them into view sequences for subsequent review or presentation. You can then play back the view sequence either interactively or by time delay.

### Editor

To define a view sequence with the View Editor, use the following steps:

1. Select View > View Sequence > Editor. The View Sequence Editor dialog box displays.




1. Click New and a text box displays.
2. Enter a name in the text box.
3. Click OK to save the name. The name displays in the View Sequence list and the existing views display in the View List.
4. Select the view name in the View List and click the Left Arrow button to move views from the View List to the Member Views list.
5. Click Apply to save your changes.

To display a view, select a view name and click the View button.

To display a note for a view select a view name and click the Note button to the right of the view button. The note displays in the View Note dialog box.

To rearrange the views, use the following steps:

1. Click the View Sequence name you want to rearrange.
2. Select the view in the Member Views list and click one of the following buttons located below the Member Views list:
   - Up Arrow: Moves the view up a level in the Member Views list
   - Down Arrow: Moves a view down a level in the Member Views list
   - Right Arrow: Moves a view from the Member Views list back to the View List
3. Click Apply after the views are in the proper order.

To add to a view sequence, use the following steps:

1. Click the View Sequence name you want to edit.
2. Select a view name from the View List and click the Left Arrow button under the View List.

   The view name is added to the list of Member Views and is removed from the View List.
3. Click Apply to save your changes.

To edit a view sequence name, use the following steps:

1. Click the name in the View Sequence list.
2. Click Edit Name and a text box displays the selected name.
3. Click OK after you edit the name.

To delete a view sequence, use the following steps:

1. Click the name in the View Sequence list.
2. Click Delete and the view sequence is removed from the View Sequence list and temporarily stored.

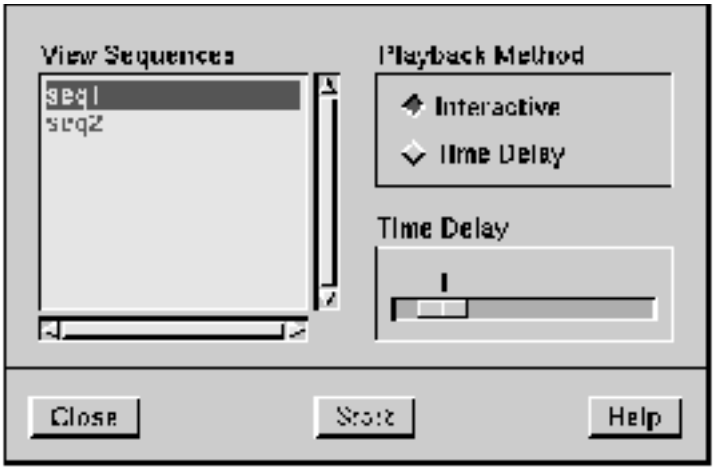
To restore a deleted view sequence, click Undo Delete to restore the view sequence. You cannot restore the deleted view sequence after you close the dialog box, define a new view sequence, or edit a view sequence name.

## Playback

After you define a view sequence, you can play back a view sequence either interactively or by time delay. If you playback a view sequence interactively, you can control when each view is displayed, as well as read the associated note. If you playback a view sequence by time delay, you can define a simulated animation sequence, for example, by adjusting the number of seconds between each view.

Before you start view sequence playback, perform the following steps:

1. Define the view sequence with the View Sequence > Editor.
2. Select View > View Sequence > Playback. The following figure shows the View Sequence Playback dialog box.




3. Set the Playback Method:
   - Interactive — Manually play back the view sequence.
   - Time Delay — Automatically play back the view sequence with the Time Delay slider scale to set the number of seconds to delay between each view.

To display a view sequence interactively, use the following steps:

1. Click a view sequence name.
2. Click Start. The first view listed in the view sequence and the following message display:

```
View sequence playback in progress...
```

3. You are prompted to choose from the following options:
   - Next View: Displays the next view.
   - Read Note: Displays the associated note. After you read a note, you must close the note window before you can display another view.
   - Stop: Stop the playback.

To display a view sequence by time delay, use the following steps:

1. Click a view sequence name.
2. Click Start. The views begin to display, along with the following message:

```
View sequence playback in progress...
```

   The View Sequence playback slider scale indicates which view is displaying.

3. Click Stop to stop the playback.

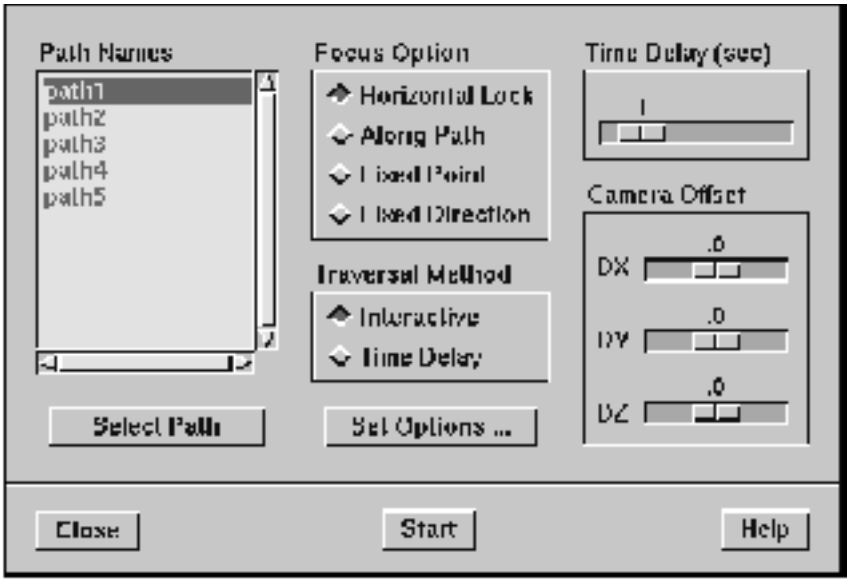
# Camera Traversal

This option allows you to traverse the camera along predefined paths through the 3D scene, while maintaining the focus of the camera in a horizontal direction, along a path, on a fixed point, or in a fixed direction. Paths display as lines within the 3D scene. Paths can be defined with the Path Editor located in the Edit menu. To create a textual graphic file that contains a path definition, refer to Appendix B, “EPD.Visualizer Textual Graphic File”.

## Before You Start

Before you start camera traversal, perform the following steps:

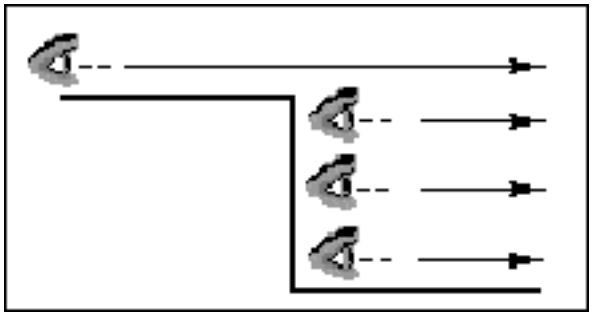
1. Add graphic file(s) with path definitions.
2. Select View > Camera Traversal. The following figure shows the Camera Traversal dialog box.



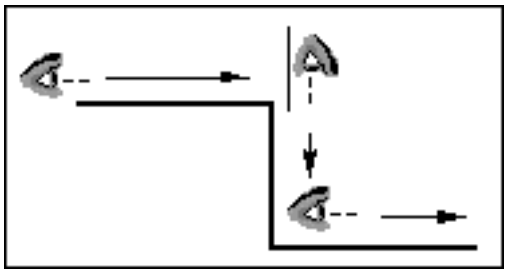
3. Set the following options as necessary:

   Focus Option

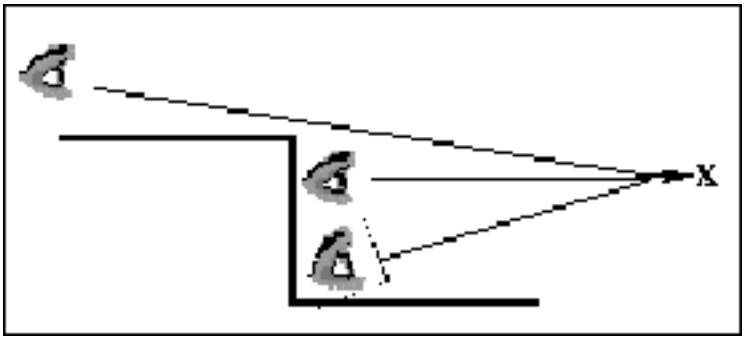
   - Horizontal Lock: The focus of the camera maintains a horizontal orientation regardless of the direction the path takes.



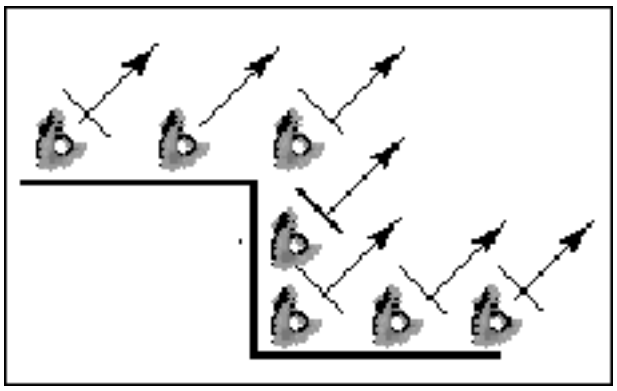
   - Along Path: The focus of the camera remains on the path. The focus maintains a horizontal orientation when the path direction is horizontal; it switches to a vertical orientation when the path direction is vertical.



   - Fixed Point: The focus of the camera remains on a fixed point.



- Fixed Direction: The focus of the camera remains on a fixed direction.

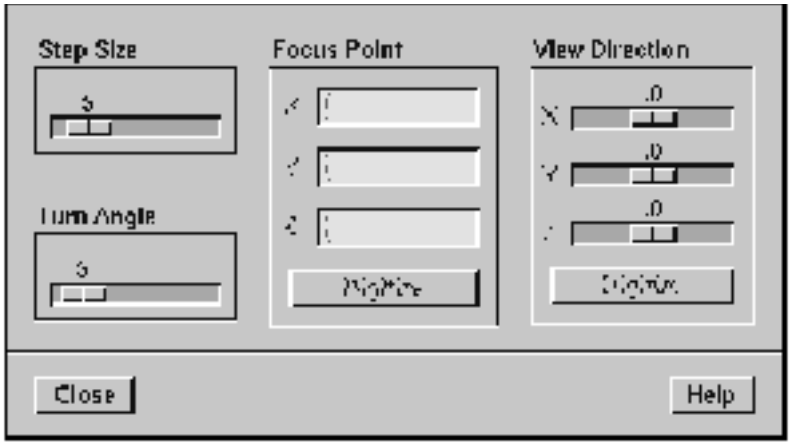


In the Traversal Method field set the following:

- Interactive: Manually step along the path. At any time, you can reverse direction and view your previous position on the path.
- Time Delay: Automatically traverse the path with the Time Delay slider scale to set the number of seconds to delay between each step. If the Movie Recorder is turned on with this method, movie images are generated for the traversal sequences.

Camera Offset — Set the offset of the camera position as it traverses along a path by setting the DX, DY, and DZ slider scales.

**a.** Click Set Options and the Camera Traversal Options dialog box displays.



**b.** Set the following options as necessary:

Step Size — Set the step size the camera takes along the path by setting the Step Size slider scale. A zero value causes the camera to take one step for each path segment.

Turn Angle (deg) — If you selected either the Horizontal Lock or Along Path options, set the angle by which the camera incrementally turns when it reaches the corners along a path by setting the Turn Angle slider scale. A zero value causes the camera to turn each path corner in one step.

Focus Point — If you selected the Fixed Point option, set a fixed focus point for the camera either by entering the xyz coordinates in the text boxes or with the Digitize button to digitize a focus point.

View Direction — If you selected the Fixed Direction option, set a fixed view direction for the camera either with the three View Direction slider scales to set the xyz values or with the Digitize button to digitize a view direction. The scales range from -1.0 to 1.0.

## Start Camera Traversal

To start camera traversal, use the following steps:

1. Select a path by one of the following methods:
   - Pick up the Select Path tool and select a path in your 3D scene
   - Click the path name from the Path Names list
   - Click Start and a dialog box displays the following message:

     ```
     Camera traversal in progress...
     ```
2. If you selected interactive camera traversal, you can either:
   - Click Next View to display the next position on the path
   - Click Previous View to display the previous position on the path

## Verify the Path Name

To verify the name of a path pick up the Select Path tool and select the path in the 3D scene. The name is highlighted in the Path Name list.

## Blend Colors

If camera traversal is done with the Time Delay method and the Movie Recorder is turned on, movie color blending is done automatically. If the traversal is canceled, you have the option to continue with the blending.

Color blending averages the colors used in the movie frames and creates a new color table containing 256 of the most used and averaged colors. It will rewrite the movie frames to use this new color table.

Please note: You must blend the colors of the movie files before you can view them with the Movie Player. Color blending could take several minutes depending on the size of the file and the length of the movie.

To blend colors for a movie, use the following steps:

1. Click Blend and a dialog box displays the following message:

```
Movie frame color blending is in progress...
```

2. Click Close when color blending is complete.

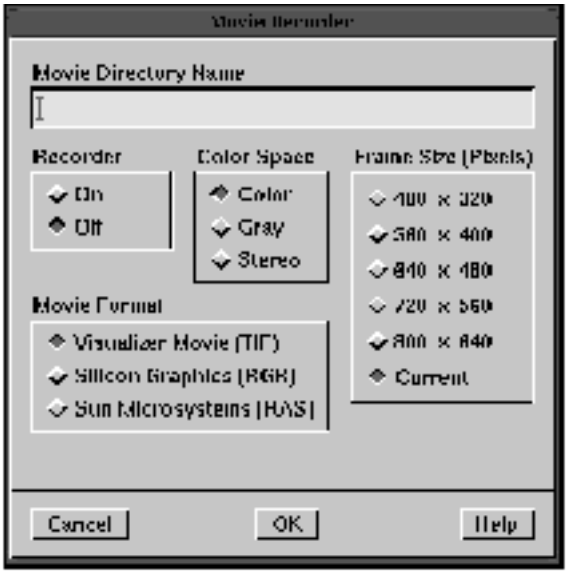
## Movie Recorder

Use this option to capture sequences of images to disk files during object and camera traversals. The purpose of recording images is to enable you to view these sequences at a much higher speed than real time rendering. With a movie player, you can view image files with CV Animator, pvsmovie or any other desktop movie viewer that can read TIF, RGB, and Sun Raster files. All recorded movie images have the extensions `.tif`, `.rgb` or `.ras`, and all the movie files have the extension `.mv`. The Movie Recorder dialog box contains components that let you define specific features of a movie.

Please note: On Sun platforms running `xgl`, do not allow any window or menu to occlude the EPD.Visualizer window while performing the capture operation. Otherwise, the output raster file will not be created properly.

To Define Movie Features, use the following steps:

1. Select View > Movie Recorder. The Movie Recorder dialog box displays:




2. Enter a movie directory (any valid directory name) in the Movie Directory Name text box.
3. Set the following options as necessary:

   Recorder — Turns recording on or off. When on, the image frames are recorded to the movie directory during object and camera traversals.

   Color Space

   - Color — Record movie in full color.
   - Gray — Record movie in gray scale.
   - Stereo — Produce stereoscopic animations for viewing through special 3D glasses. Output is through movie creation as 8-bit TIFF files in your movie directory.

   Frame Size

   - Click the desired frame size and the window resizes to the selected dimension.
   - Current — Use the current graphics window size.

   Movie Format

   - Visualizer Movie (TIF)
   - Silicon Graphics (RGB)
   - Sun Microsystems (RAS)

4. Click OK and the following message displays:

```
Movie recording is now enabled for Camera and Object
traversal
```

This message also displays the movie directory and the starting frame number. After the movie recorder is turned on, you can start recording movies with Camera or Object Traversal. Refer to “Camera Traversal” on page 3-66 and “Path Traversal” on page 3-16 for more information.

# Options Menu

By selecting Options on the menu bar, you can customize your working environment with the following menu items:

- Hide/Show Palette — Hide or show the palettes
- Left/Right Palette — Move the palettes to the left or right side of the graphics window
- Preferences — Set working units, graphics window color, and mouse dynamics
- Temp Color — Set the color of temporary graphics and secondary view boxes
- Axis Display — Switch display of axis on/off

## Hide/Show Palette

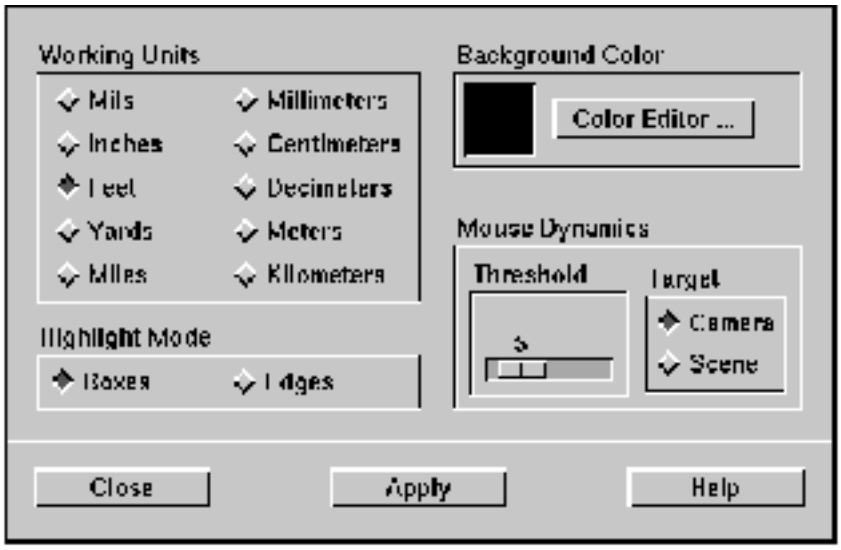
To hide or show the palettes, select Options > Hide/Show Palette.

## Left/Right Palette

To move the palettes to the left or right side of the graphics window, select Options > Left/Right Palette.

# Preferences

To change the working units for the 3D scene, the color of the graphics window, or the behavior of the mouse dynamics, select Options > Preferences. The User Preferences dialog box displays.




- Working Units —To set the working units select the appropriate working unit, click Close and the working unit displays wherever units display.
- Highlight Mode — To set the highlighting mode select one of the following modes. The default is boxes.

Please note: Boxes highlighting mode executes much faster then Edges mode when selecting a large number of items.

  - Boxes — Highlight selected objects with their extents boxes.
  - Edges — Highlight selected objects with their polygon edges.
- Background Color — To set the color of the graphics window, select the color in the Background Color Editor. To use the Color Editor, refer to “Color Editor” on page 1-19.

- Mouse Dynamics — The mouse dynamics option enables you to set the threshold and target for the Mouse Viewing tool. (For more information on this tool, refer to “Mouse Viewing” on page 9-9.)

  Set the mouse dynamics with the following options:

  - Set the mouse threshold with the slider scale. Threshold refers to the number of pixels the mouse pointer travels before the display is updated. Threshold levels can range from 2 to 25 pixels. The default is 5.
  - Select the target. You can select either camera or scene. The default is camera.

    — Camera: Moves the camera position in the direction of the mouse pointer.

    — Scene: Moves the 3D scene in the direction of the mouse pointer. The camera remains stationary. You can enable scene mode when starting EPD.Visualizer using the `-scene` option in the EPD.Visualizer command line. Refer to installation documentation for more information.

## Temp Color

By default, temporary graphics are displayed in black or white, depending on your background display. You can specify a color for EPD.Visualizer to use when displaying temporary graphics such as selection and digitize markers, as well as secondary view orientation boxes.

To specify a color for temporary graphics, use the following steps:

1. Define the environment variable `PVS_TEMP_COLOR`, as in the following syntax:

   ```
   setenv PVS_TEMP_COLOR green
   ```

2. Select Options > Temp Color > User.

To revert to the system default, select Options > Temp Color > Standard.

# Tools Menu

The Tools menu contains selections that perform the functions represented by the icons in the tools palette section of the EPD.Visualizer window. This menu gives you the option of the tool capability without opening the tool palette.

In addition to the tool palette, the Tools menu contains selections that interact with the product structure tree.

This section briefly describes the Tools menu commands. For a complete description of the functions associated with the EPD.Visualizer tools palette, refer to Chapter 9, “Palettes.”

The following sections explain the tools that are available.

## Tree

The Tools > Tree command enables you to define how graphic components are selected.

- Select & Scroll — Select this command, then select a graphic component. In the EPD.Connect window, the product structure tree scrolls to the items associated with the selected component. A Tree Select and Scroll tool is also available in the Palette.

  You can run custom reports by using the Select and Scroll option. Set the values of the variables `PVS_SELECT_SCROLL_REP` and `CA_ENV_REPORT`. See “Selecting, Scrolling, and Running a Report” on page H-12 for details.
- Select — Select this command to select the graphic component and the tree items associated with the component.

Please note: To select more than one graphic component, hold down the Shift key while making your selections.

- Scroll — Scrolls the tree to the tree components associated with graphic components selected in EPD.Visualizer.
- Deselect — Clears one graphically selected component from the EPD.Visualizer selection list and from the product structure tree.

Please note: The Tree menu commands listed thus far remain selected until another tool or command is selected.

- Deselect All — Clears all product structure tree components and all EPD.Visualizer graphic components.

- Highlight Visible — Highlights in the product structure tree all components for which any part of their associated graphics are visible in EPD.Visualizer.
- Select Selected — Selects in the product structure tree all components that have any of their graphics selected in EPD.Visualizer.

## Item

The Tools > Item command enables you to select and verify items.

- Select — Select single or multiple items by group or discipline, as defined by the Select > Selection Masks command.
- Verify — Retrieves object information about a selected item.

## Mouse Dynamics

The Tools > Mouse Dynamics command enables you to change your view orientation with the mouse. For complete information about this feature, refer to “Mouse Viewing” on page 9-9.

## Zoom

The Tools> Zoom command enables you to define a rectangular area to be a temporary or permanent close-up view of your 3D scene. For more information about this feature, refer to “Zoom” on page 9-6.

## Set Camera

The Tools > Set Camera command provides the following selections that enable you to manipulate the camera:

- Position — Set a new position (eye point) for the camera. Refer to “Set Camera Position” on page 9-6 for further information on position
- Target — Set a point around which the orbit buttons operate. Refer to “Set Camera Target” on page 9-7 for further information on target.
- Position and Target — Set a new camera position and camera target. Refer to “Set Camera Position” on page 9-6 and “Set Camera Target” on page 9-7 for further information.

## Measure Distance

The Tools > Measure Distance command measures the distance between the camera position and a vertex point you select in the 3D scene, or the distance between two vertices. It also measures distances in sections. Refer to Chapter 7, "Controlling a Plane's Display and Multiple Sectioning" for information about measuring distances in sections.

## Zone Definition

The Tools > Zone Definition command allows you to construct three-dimensional zones for circumscribing areas for clash detection. Refer to Chapter 4, "Constructing Zones" for detailed information about constructing zones.

## Normals

You can display normals to a selection or surface and then erase the normals with the following selections. The normals' lengths are determined automatically from the visible parts of the scene. A gap of 10% of each normal's length separates it from its surface.

Surfaces in EPD.Visualizer are constructed from polygons. The normals are generated from locations stored as the first points of polygons, so if polygons share the same first point, the normals from them can merge, which gives the appearance of a single normal.

- Tools > Normals > Selection — This operation displays a set of normals to the current selection. You must select the function each time you use it.
- Tools > Normals > Point — This operation displays a normal to the surface for each point that you select by clicking. You can select any number of points, one at a time.
- Tools > Normals > Clear — This operation erases normals in the graphics window.

# Select Menu

With the Select menu, you can manage items by adding them to a selection set or selection list with the Select tool or the following menu items:

- Selection Masks — Set selection mask for digitized selections
- Selection Set Editor — Create and manage selection sets
- Build Selection List — Build selection list consisting of items from Graphic Files, Selection Sets, Disciplines, or Groups
- Copy Sel. -> Sel. 2 — Create a secondary selection by copying the current selection to it
- Deselect All — Clear all items
- Clear Selection 2 — Clear the secondary selection
- Highlight Selection — Turn on or off highlighting of selected items

You can create selection sets consisting of items that you add to a selection list with either the Select tool or by selecting Build Selection List. You can save selection sets with a metafile.

## Selection Masks

Before you create a selection list, set the selection mask to specify graphics files, disciplines, groups, or items by selecting Select > Selection Masks. After you select an item in your 3D scene, all masked items are added to the selection list.

You can set one of the following options. The default setting is Graphics File.

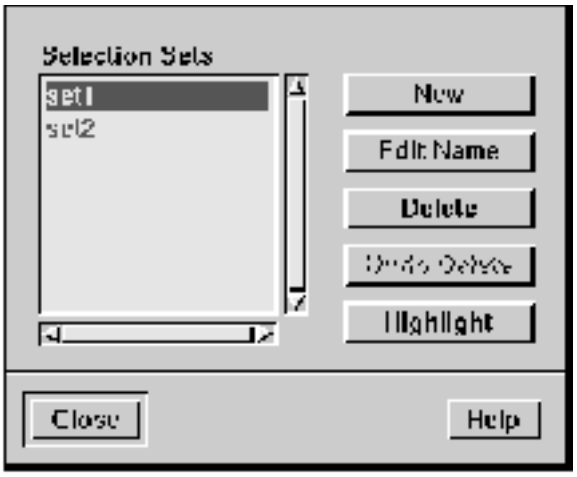
- Graphics File — After you select an object in your 3D scene, the graphics file is added to the selection list.
- Disciplines — After you select an object in the 3D scene, all objects within the same discipline are added to the selection list. For example, if you select a piece of pipe, all piping components belonging to the same discipline are added to the selection list.
- Groups — After you select an object in the 3D scene, all objects in the same group are added to the selection list.
- Item — After you select an item in your 3D scene, only that item is added to the selection list.

# Selection Set Editor

A selection set is a named collection of items that you can save and later add to the selection list. You can save selection sets with a metafile. After the selection set is defined, you can add it to the selection list and highlight it by selecting Select > Build Selection List.

To define a selection set, use the following steps:

1. Add item(s) to the selection list.
2. Select Select > Selection Set Editor and the Selection Set Editor dialog box displays.




3. Click New and a text box displays.
4. Enter a name in the text box.
5. Click OK to save the name and it displays in the Selection Sets list.

To edit a selection set name, use the following steps:

1. Click the name in the Selection Sets list.
2. Click Edit Name and a text box displays the selected name.
3. Click OK after you edit the name.

To delete a selection set, use the following steps:

1. Click the name in the Selection Sets list.
2. Click Delete and the selection set is removed from the Selection Sets list and temporarily stored.

To restore a deleted selection set click Undo Delete. You cannot restore the deleted selection set after you close the dialog box, define a new selection set, or edit a selection set name.

To highlight a selection set, use the following steps:

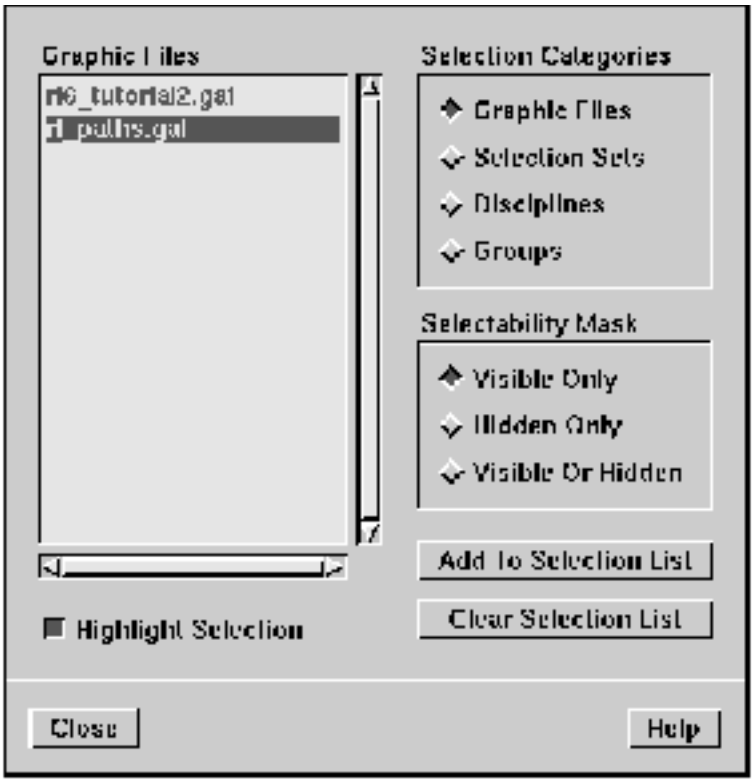
1. Click the name in the Selection Sets list.
2. Click the Highlight button and the items in the selection set are highlighted.

## Build Selection List

You can build selection list that consist of items from Graphic Files, Selection Sets, Disciplines, or Groups. You can add any number of items to the selection list.

To build a selection list, use the following steps:

1. Select Select > Build Selection List. The Build Selection List dialog box displays.



2. Set the following options as necessary:

   Selection Categories — Click one of the following categories and a listing of that category displays to the left of the Selection Categories box:

   - Graphic Files — All currently added graphic files
   - Selection Sets — All existing selection sets
   - Disciplines — All disciplines defined in the graphic files
   - Groups — All groups defined in the graphic files

   Highlight Selection — Turn Highlight Selection on or off, depending on whether you want to highlight items on the selection list. To highlight items on the selection list, click the button on.

   Selectability Mask — Choose one of the following options:

   - Visible Only — Only visible item(s) are added to the selection list
   - Hidden Only — Only hidden item(s) are added to the selection list
   - Visible Or Hidden — Visible and hidden item(s) are added to the selection list

To add item(s) to the selection list, select the entry or entries in the listing. Then click Add To Selection List.

Please note: To select multiple entries, hold down the Control key and click random entries.

To clear the selection list, click Clear Selection List.

## Copy Selection to Secondary Selection

Select Select > Copy Sel. -> Sel. 2 to create a secondary selection by copying the current selection to it. This is useful for both static and dynamic interference checking, as described in Chapter 5, “Interference Checking.” Sectioning and multiple sectioning from Gpath references can operate on the secondary selection.

## Deselect All

If you select Select > Deselect All, all items are cleared and the selection list is cleared.

## Clear Secondary Selection

Select Select > Clear Selection 2 to clear the secondary selection you created with Select > Copy Sel. -> Sel. 2, as described previously.

## Highlight Selection

With this option switch, you can turn the highlighting of items on the selection list on or off.

Please note: The Highlight Selection switch is also located on the Translate/Rotate Selection List dialog box and Build Selection List dialog box. If you turn one of these switches on or off, all of the switches are affected.

# Database Menu

The Visualizer release 1.1.1 has an SQL interface that lets you connect to the Oracle Database Management System (DBMS) in addition to opening an ASCII database file. You connect and disconnect from the DBMS through the Database menu. Only one DBMS or ASCII file can be active at any time.

You can use a database for the following purposes:

- Obtain an object's engineering data, with the Verify tool.
- Tag an object with the tag stored in the database file by selecting Database > Tagging > Automatic and then the Object Tagging tool.
- Send SQL statements to edit or select data from DBMS tables. Objects selected from the DBMS can be highlighted, zoomed or added to the selection list.

The following menu items are available within the Database menu:

- Connect — Open an ASCII database file or connect to a DBMS
- Disconnect — Close an ASCII database file or disconnect from a DBMS
- Query Types — Specify the type of data to be retrieved
- Selection Keys — Define key field names for object selection
- Substitution Variables — Define substitution variables used in query statements
- Query Database — Send an SQL statement to the connecting DBMS for selecting data
- Tagging — Set object tagging to manual or automatic

## Connect

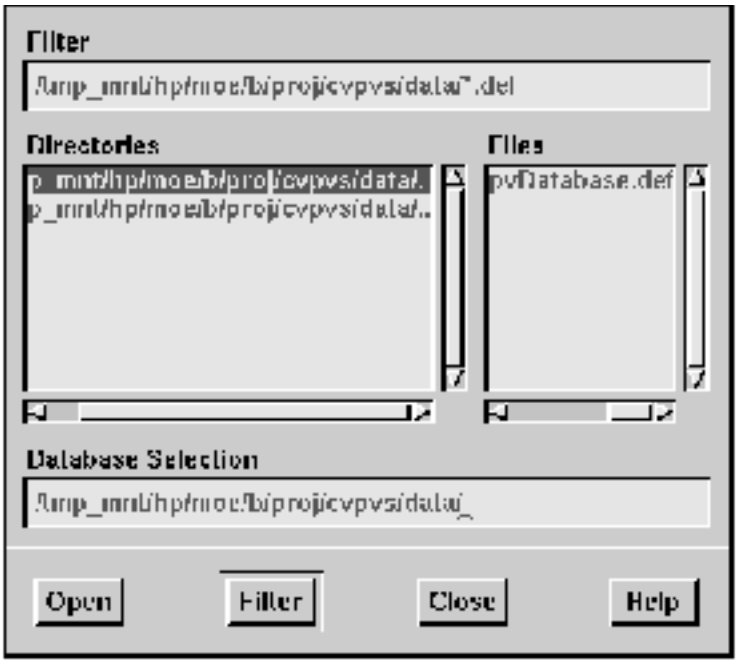
You connect to the DBMS through the Database menu. Only one ASCII or DBMS file can be active at any time.

### ASCII

An ASCII database can have an unlimited number of records. Refer to Appendix A, "Database Interface" for information on how to create a database.

To open an ASCII database, use the following steps:

1. Select Database > Connect. Select ASCII, the Flat Database File Selection dialog box displays:




2. Select a database definition file name to open from the Files list box. The default extension for database definition files is `.def`.

   Refer to "File Selection Dialog Box" on page 1-17 for more information on the filter option.

3. Click Open to open the database. The following message displays:

   `Database was opened and contains` number `records.`

   You can now access the database information when you use the Verify or Object Tagging tools.

### DBMS

To connect to a DBMS, use the following steps:

1. Select Database > Connect > DBMS. A Database Connection dialog box displays as shown below:

   Enter a valid user ID and password.

2. Click Connect.

   If the connection is not successful, the following message displays:

   ```
   Database connection failed!
   ```

   This condition is often caused by invalid user ID and/or password, invalid database name, or the Oracle or DBMS daemon not running.

   If the connection is successful, you can perform subsequent database operations such as object verification, tagging, and database query.

## Disconnect

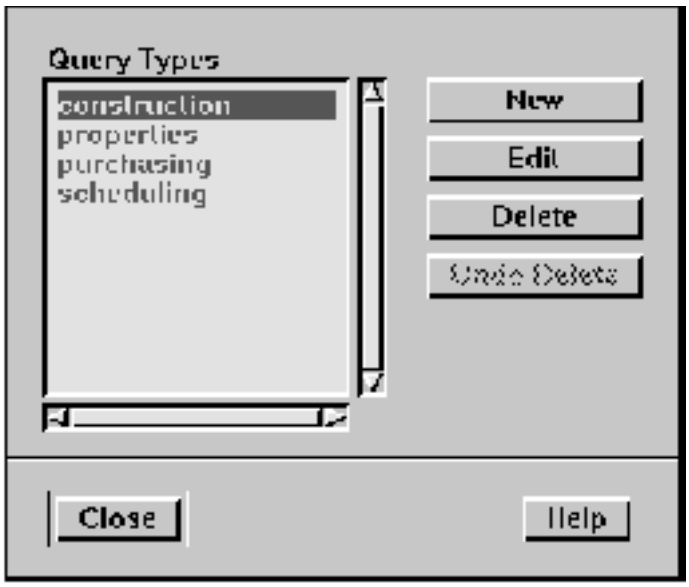
To disconnect from the database, select Database > Disconnect.

## Query Types

Query types specify the type of data to be retrieved by the Verify tool. You can define query types in a `GAF` file or with interactive commands. For more information on `.gaf` format, refer to Appendix B, “EPD.Visualizer Textual Graphic File”.

To define a new query type, use the following steps:

1. Select Database > Query Types. The Query Types dialog box displays.




2. Click New. The Define Query Type dialog box displays.
3. Enter a query type name.
4. Click OK. A new query type is defined.

To edit a query, type click Edit. The Edit Query Type dialog box displays the selected query type. Click OK after you edit the query type.

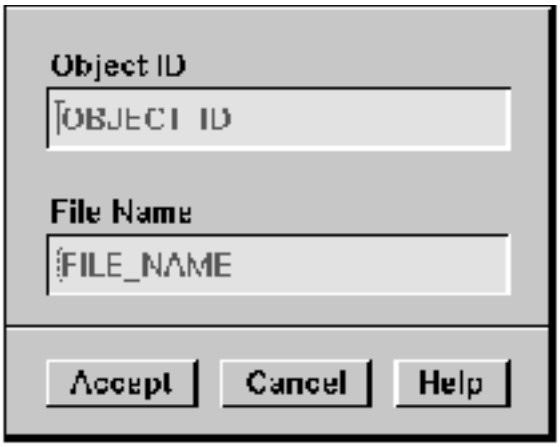
To delete a query type, click the query type you want to delete from the list. Then click Delete and the query type is removed from the Query Types list and temporarily stored.

To undelete a query type, click Undo Delete and the query type is restored to the list. You cannot restore the deleted query type after you define a new query type or edit a query type name.

## Selection Keys

If you want to highlight, zoom or select objects retrieved from the database, you must specify the Object ID and File Name in the field name list of the query statement.

The Selection Keys dialog box lets you tailor the actual field names of these two keys.




To tailor the two key fields, use the following steps:

1. Enter a field name for Object ID.
2. Enter a field name for File Name.
3. Click Accept.

   The default values are `OBJECT_ID` and `FILE_NAME`.

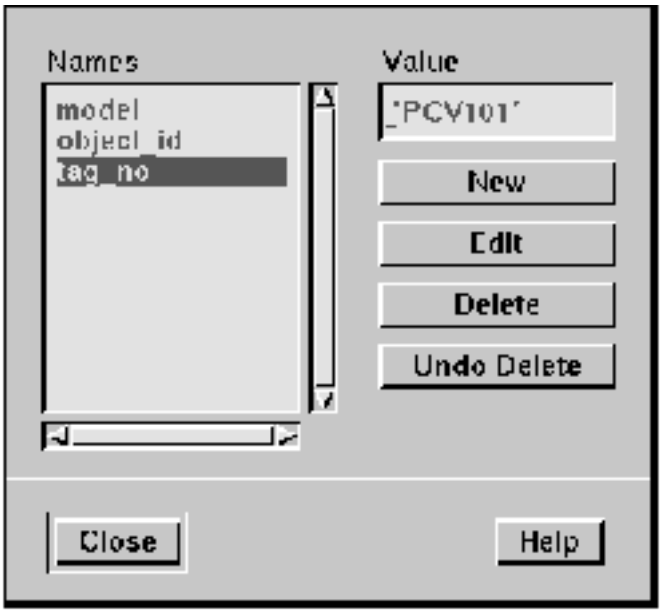
   Click Cancel to restore previously defined field names.

## Substitution Variables

Substitution Variables are used in query statements to globally set values for query variables. Variables created with this option are considered persistent variables. For example, a project name is a variable that is used throughout the querying session, and is likely to stay constant, so you might want to define that name as a persistent variable.

To define a new substitution variable, use the following steps:

1. Select Database > Substitution Variables. The Substitution Variable dialog box displays.




2. Click New.
3. Enter a name and value.
4. Click OK.

To edit a substitution variable, use the following steps:

5. Click the name of the variable to be edited in the Names list.
6. Click Edit.
7. Click OK after you edit the name and value.

To delete a substitution variable, click the name of the variable to be deleted. Then click Delete and the name is removed from the Names list.

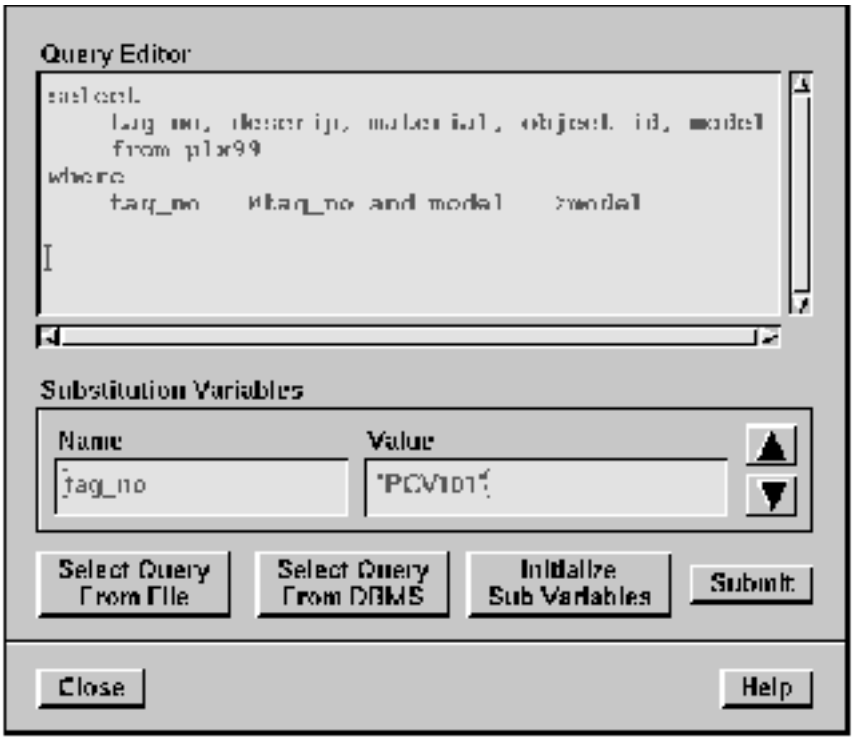
To restore a deleted substitution variable, click Undo Delete. You cannot restore the deleted variable after you define a new variable or edit a variable.

# Query Database

The DBMS Interface enables you send SQL query statements to the connecting DBMS database. The SQL query statements can be pre-defined in a file, in the connecting DBMS or entered interactively.

To create or edit an SQL query statement, use the following steps:

1. Select Database > Query Database. A Query Database dialog box displays:




2. Set the following options as necessary:

   Query Editor — Contains the contents of the SQL statement. You can enter the query interactively, load it from a previously created query file or load it from the connecting DBMS.

   Substitution Variables — A substitution variable in a query statement is denoted by a preceding '@' character. When a query statement is read in from a file or the DBMS, the Substitution Variables box is initialized with a list of variable names and their default values. If a variable is defined as a persistent variable, it is initialized to the previously defined value. Other variables obtained from the query statements are transient variables. Transient variables are deleted after the submission of the query.

   - Name — Variable name.
   - Value — Variable value. Values for string variables must be enclosed in single quotes.

You can use the up and down arrows to move between variables, or you can enter a <CR> from the Substitution Value box to move to the next variable.

Select Query From File — Load a query statement from a previously created query file. Refer to "Select a Query from a File" on page 3-91 for more information.

Select Query From DBMS — Load a query statement from the `PVS_SQL_STMT` table of the connecting DBMS. Refer to "Select a Query From DBMS" on page 3-92 for more information.

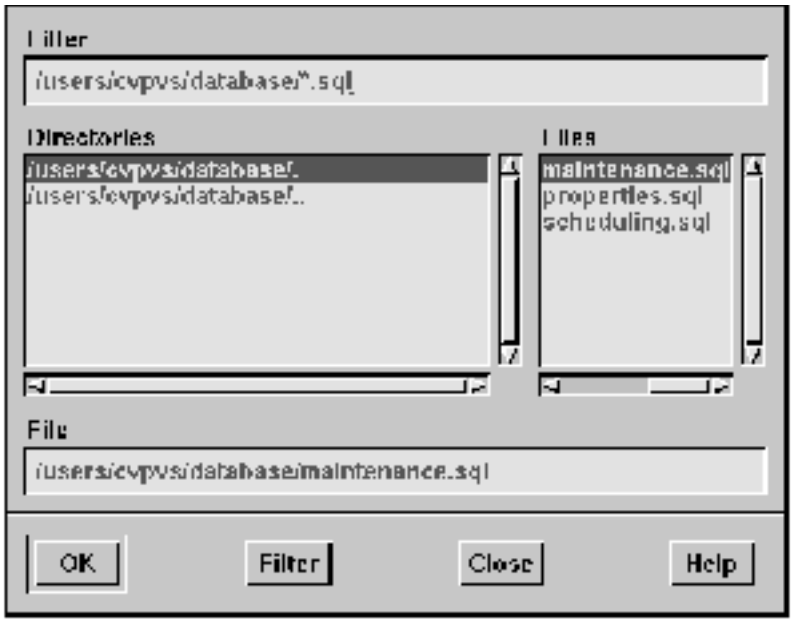
Initialize Sub Variables — Scans the contents of the SQL statement in the Query Editor and initializes the names and values of Substitution Variables box to the new variables.

Submit — Submit the contents of the Query Editor to the connecting DBMS. If this statement returns any data, the Query Database Data dialog displays. Refer to "Query Database Data" on page 3-93 for more information.

## Select a Query from a File

To select a query from a file, use the following steps:

1. Click Select Query From File. The following figure shows the Select Query File dialog box.




2. To select an existing file, click the file name in the Files list and the file name displays in the File text box. The default extension for the SQL files is `.sql`.

   Refer to "File Selection Dialog Box" on page 1-17 for more information on the filter option.

3. Click OK.

   The contents of the file displays in the Query Editor. Extract and list the names of the substitution variables in the Substitution Variables box.

## Select a Query From DBMS

The DBMS Interface lets you send SQL queries statements to the connecting DBMS database. The SQL query statements can be pre-defined in a file, in the connecting DBMS or entered interactively in.

To select a query from DBMS, use the following steps:

1. Click Select Query From DBMS. The Select Query From DBMS dialog box displays.

2. Select a query name you want from the list.

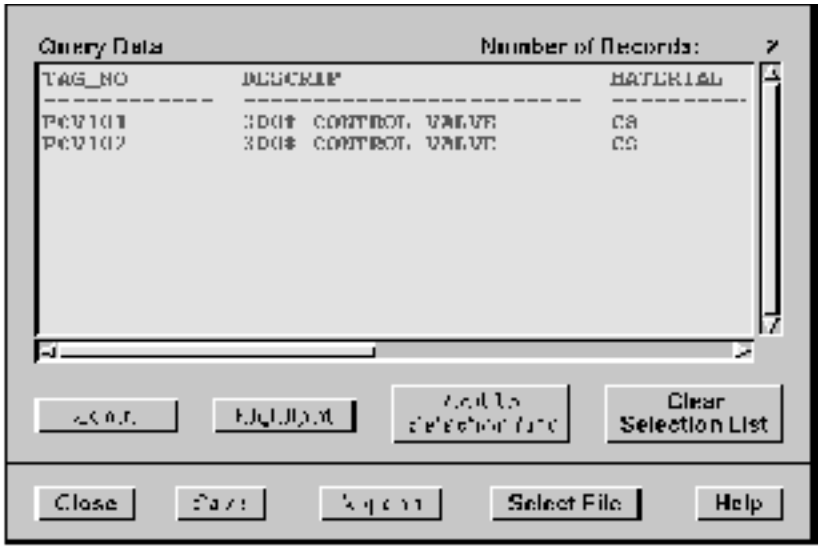
   Query names are obtained from the `PVS_SQL_STMT` table of the connecting DBMS. Refer to Appendix A, “Database Interface” for more information on the format of this table.

3. Click Retrieve.

4. The query statement corresponding to the name selected is retrieved and displayed in the Query Editor. The names of the substitution variables are extracted and listed in the Variable Substitution box.

## Query Database Data

The following Query Database Data dialog box displays if any data is returned by the SQL statement.



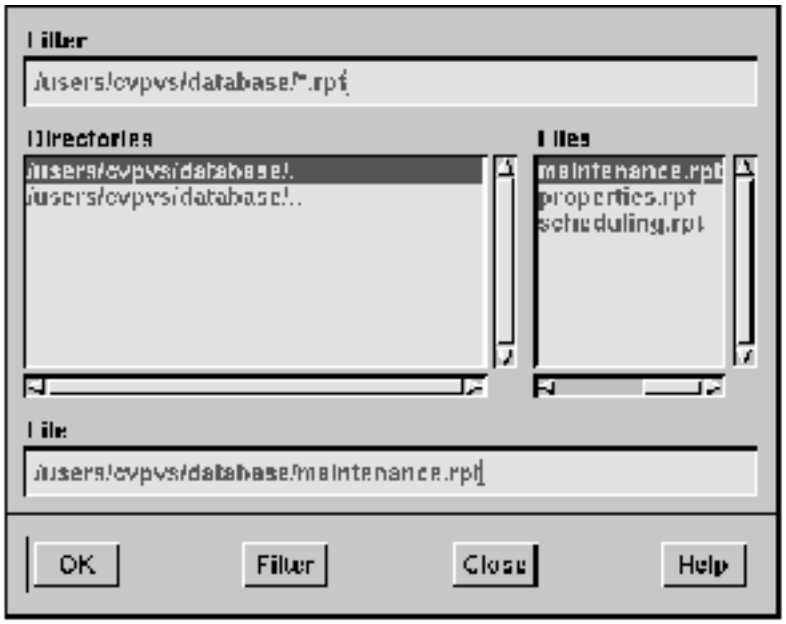

The following text explains the fields and buttons on the Query Database Data dialog box.

- Query Data — A scrolled list containing all the records returned by the query statement.
- Zoom — Zooms to the extents of the object(s) selected in the data list.
- Highlight— Highlights the object(s) selected in the data list.
- Add to Selection List — Adds the object(s) selected in the data list to the selection list.
- Clear Selection List — Clears the selection list.

## Save a DBMS Report

To save a DBMS report, use the following steps:

1. Click Select File. The following figure shows the DBMS Report File Selection dialog box.




2. Click the file name in the Files list to select an existing file, and the file name displays in the File text box.

   Enter the directory path and/or new file name in the File text box to create a new file.

   Use a `.rpt` extension following the file name to organize your files.

   Refer to "File Selection Dialog Box" on page 1-17 for more information on the filter option.

3. Click OK.

4. Click Save and the DBMS data displayed in the Query Data list is written to the file you selected or named.

Please note: You can overwrite an existing file name by entering the same name in the File text box.

5. Click Append to append the report to the end of the file you selected or named.

Any additional Save or Append commands access the current file until you click Select File to select or name another file.

## Manual or Automatic Tagging

You can either select automatic object tagging, which is based on a tag defined in the database, or you can manually tag objects when you use the Object Tagging tool. Refer to “Tool Palette” on page 9-2 for an explanation of the Object Tagging tool.

# Interference Checking Menu

Interference Checking allows you to perform interference checking between objects in the selection list and surrounding objects. By selecting Interference on the menu bar, you can access the following menu items:

- Item Dynamics — Turn interference checking on or off for item editing or item traversal operations. The status of this item automatically synchronizes with the following:
  - Item Dynamics in the Interference Checking - Control dialog box
  - Check Collisions in the Object Path Traversal dialog box
- Control — Set up your interference checking session
- Clear — Deactivates interference checking and restores original colors

Refer to Chapter 5, “Interference Checking” for detailed information about interference checking.

# Sectioning Menu

Sectioning allows you to scan an assembly by defining and moving a plane. By selecting Sectioning on the menu bar, you can access the following menu items:

- Plane Dynamic — Set a sectioning plane to be dynamic
- Control — Specify the display and behavior of a sectioning plane
- Clear — Deactivate clipping and sectioning

Refer to Chapter 6, “Sectioning”; Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning”; and Chapter 8, “Moving a Plane” for detailed information about clipping and sectioning.

# Online Help Menu

You can access the EPD.Visualizer online help to obtain immediate assistance. By clicking on Help in the menu bar, you can access the following menu items:

- On Help — Describes how to use context-sensitive help and other options
- On Context — Provides online help for the menu bar and palettes
- On General — Provides basics on the EPD.Visualizer interface
- On Glossary — Provides a glossary of terms used in EPD.Visualizer

For quick reference help, you can point to the tools, buttons, or slider scales and a brief description displays in the prompt window.

## On Help

Select Help > On Help for an explanation on how to use context-sensitive help and the other help options.

## On Context

You can obtain context-sensitive help on the menu names located on the menu bar at the top of your EPD.Visualizer window and the palettes (containing tools, buttons, and slider scales) located beside your EPD.Visualizer window.

To use On Context perform the following steps:

1. Click On Context and your pointer changes to a question mark (?).
2. Click any of the menu names, tools, buttons, or slider scales, and a window displays a description on how to use the selected item. To use On Context on the slider scales in the palette, click on the slider scale value.
3. To scroll up or down the On Context window, click in the scroll bar on the right.
4. Click Close to close the window.

## On General

Select Help > On General for basics on with the EPD.Visualizer interface.

## On Glossary

Select Help > On Glossary for a glossary of terms used in EPD.Visualizer.

## On Environment

Select Help > On Environment for a list of EPD.Visualizer environment variables and directions about setting them.

## On Release Notes

Select Help > On Release Notes for a list of new features in the current EPD.Visualizer revision.

Chapter 4

# Constructing Zones

This chapter describes how to construct three-dimensional zones within an assembly. After creating a zone, you can test for physical clashes within it. The chapter includes information on the following:

- About Zones
- Environment Variables You Need to Set
- Creating Your Zone Definition

# About Zones

A zone is a three-dimensional object shaped as a box or cylinder. You can create zones to circumscribe areas within an assembly for clash detection.

You have to create zones both as EPD.Visualizer geometry and as external graphic files (`.gaf`). The color definition for any zone is red for external faces and green for internal faces. The internal parallel faces are not visible in EPD.Visualizer even if the polygons are generated both in EPD.Visualizer and in the `.gaf` file.

You are allowed to have only one zone defined in EPD.Visualizer at a time. The current zone is automatically removed when you leave the zoning environment.

Zone definition requires that you set environment variables for the directory used to store the zones you create and for the space between the internal faces. These environment variables are described in the following section.

# Environment Variables You Need to Set

To construct a box or cylinder zone, you must set two environment variables. These variables are described in the following table:

| Variable | Description |
|---|---|
| CVDPA_ZONES_DIR | Specifies the directory used to store the zones you create.<br>For example:<br>`setenv CVDPA_ZONES_DIR /usr/tmp/zones` |
| CVDPA_SPACING | Specifies the value defining the step between internal faces.<br>For example:<br>`setenv CVDPA_SPACING 50` |

# Creating Your Zone Definition

To create a zone, select Tools > Zone Definition from the EPD.Visualizer window. Two options, Box and Cylinder, appear. These options are described in the following sections.

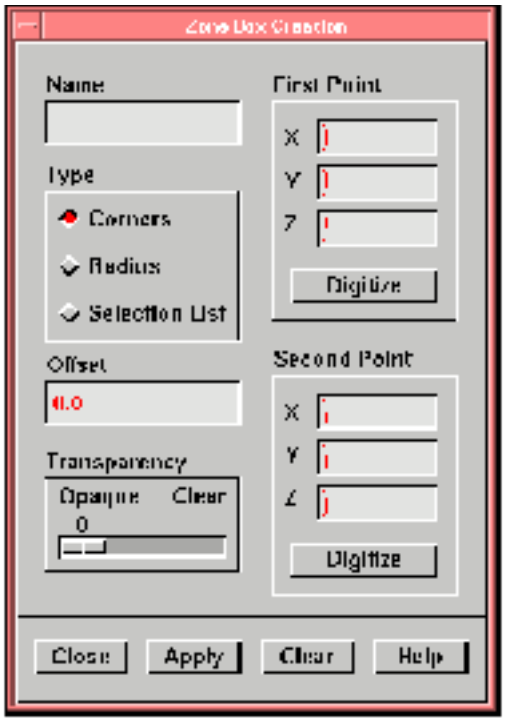
## Creating a Box Zone

A box zone is a three-dimensional box volume filled with parallel faces spaced according to the value defined for the `CVDPA_SPACING` variable. You offset and display the external faces of the box by specifying an offset value and a transparency value. Internal faces are created but not displayed. The box edges are always oriented along the main X, Y, Z axes. The extents are computed in different ways depending on the box type you select.

The box types you can create are described in the following table:

| Box Type | Description |
|---|---|
| Corner Box | Extents are computed from two points that you enter in the Zone Box Creation window. See page 4-6. |
| Radius Box | Extents are computed as extents of the inscribed sphere having the center and radius that you enter in the Zone Box Creation window. See page 4-6. |
| Selection List | Extents are computed as including all the entities contained into the current selection list. See page 4-7. |

To create a box zone, select Box from the Zone Definition menu. The Zone Box Creation window appears.




The options available in the Zone Box Creation window are described in the following table:

| Option | Description |
|---|---|
| Name | Name of the zone. This name is stored in the directory defined by the `CVDPA_ZONES_DIR` environment variable. |
| Type: Corners, Radius, Selection List | Type of box you want to create.<br>Corners is a two-point definition.<br>Radius is one point and an inscribed sphere radius.<br>Selection List requires a previously created selection. The box encloses all objects in the selection list. |
| Offset | Offset value for the external faces. |
| Transparency | Transparency value for the external faces. |
| First Point | Coordinates for the first point. (Corners type only.) |
| Second Point | Coordinates for the second point. (Corners type only.) |

You can enter explicit numerical values for the first and second points in the window's text boxes or specify them by digitizing an existing object.

The buttons at the bottom of the Zone Box Creation window are described in the following table:

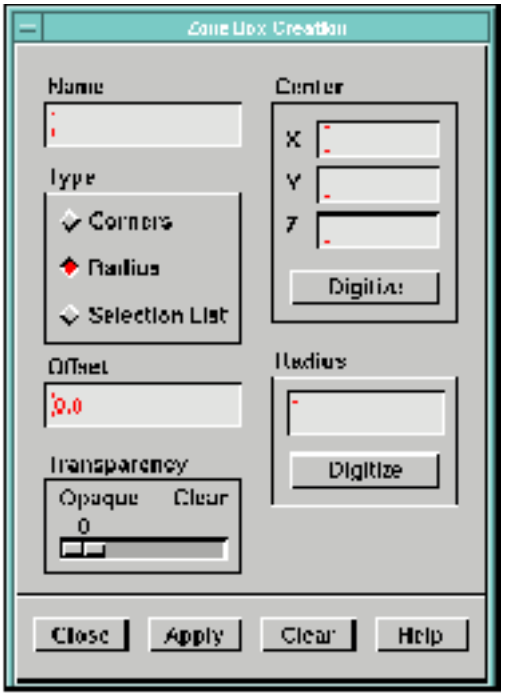
| Button | Description |
|---|---|
| Apply | Creates the zone object in EPD.Visualizer and the corresponding `.gaf` file. The previously created zone is automatically deleted from EPD.Visualizer. |
| Close | Cancels the operation and menu. The current zone geometry is automatically deleted from EPD.Visualizer. |
| Clear | Deletes the previously created zone. |
| Help | Displays online help. |

## Corners Box Type

The corners box type is selected by default in the Zone Box Creation window. The information you need to provide for a corners box is described in the previous options table.

## Radius Box Type

To create a radius box, select the Radius option from the Zone Box Creation window. The text boxes for a radius box appear.

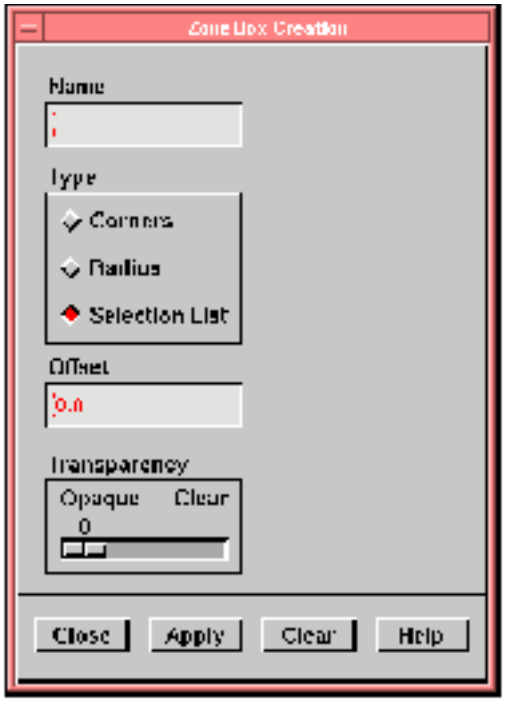



In addition to specifying offset and transparency values, as you do for a corners box, you need to enter center point coordinates and a radius. Again, you can enter

explicit numerical values for the center points in the window's text boxes or specify them by digitizing an existing object.

### Selection List

To create a box for a selection list, select the Selection List option from the Zone Box Creation window. Text boxes for offset and transparency values appear.

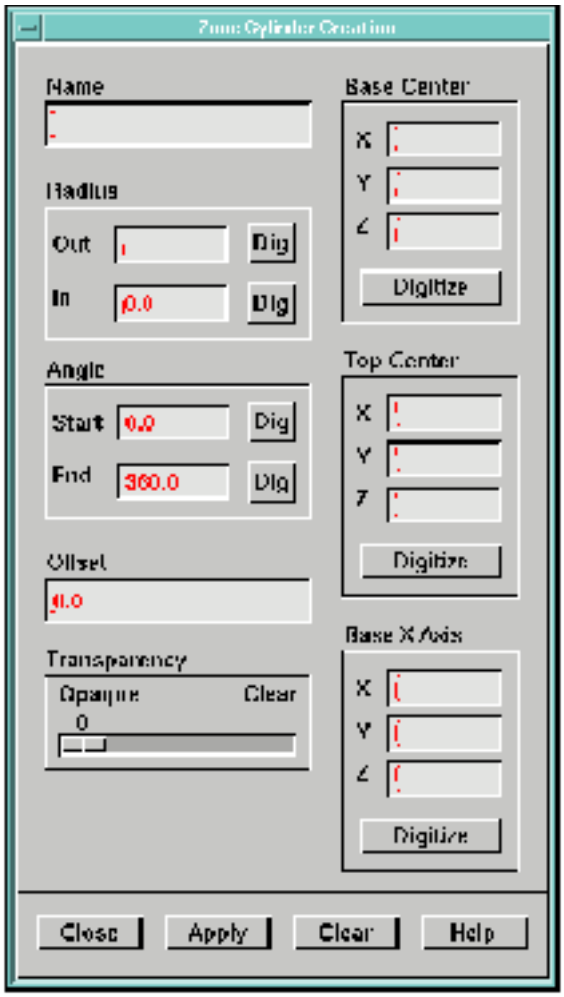



## Creating a Cylinder Zone

A cylinder zone is a three-dimensional cylindrical volume filled with parallel faces spaced according to the value defined into `CVDPA_SPACING` variable. A cylinder zone has base and top faces centered in points that you specify as well as an axis along the line defined by the two points. The circular inner and outer shapes are tessellated with a number of 20 points. This value is set with the `CVDPA_SPACING` variable.

You offset and display the external faces of the cylinder by specifying an offset value and a transparency value. The internal faces are defined in two directions, along the cylinder axis and normal to the axis itself, and along the cylinder radius. Depending on the outer/inner radius and the start/end angle you specify, your cylinder can have a cylindrical, wedge, or tubular shape.

To create a cylinder zone, select Cylinder from the Zone Definition menu. The Zone Cylinder Creation window appears.




The options available in the Zone Cylinder Creation window are described in the following table:

| Option | Description |
|---|---|
| Name | Name of the zone. This name is stored in the directory defined by the `CVDPA_ZONES_DIR` environment variable. |
| Radius: Out, In | Out Radius is the external radius.<br>In Radius is the internal radius. |
| Angle: Start, End | Start Angle is the start angle for a partial cylinder.<br>End Angle is the end angle for a partial cylinder. |
| Offset | Offset value for the external faces. |
| Transparency | Transparency value for the external faces. |

| Option | Description |
|---|---|
| Base Center | Coordinates for the base center vertice. |
| Top Center | Coordinates for the top center vertice. |
| Base X Axis | Coordinates for the base X axis. |

You can enter numerical values as explicit values in the window's text boxes or specify them by digitizing an existing object.

Chapter 5

# Interference Checking

This chapter describes interference checking and includes information on the following:

- About Interference Checking
- Starting Interference Checking
- Setting Up an Interference Checking Session
- Reviewing the Results

# About Interference Checking

Interference checking allows you to perform spatial studies to detect potential design flaws early in the engineering process before construction or fabrication begins. Interference checking allows you to do the following:

- Perform static and dynamic interference checking
- Color and highlight colliding components
- Check components against a selection
- Analyze and display colliding components
- Review the results
- Generate reports of the results

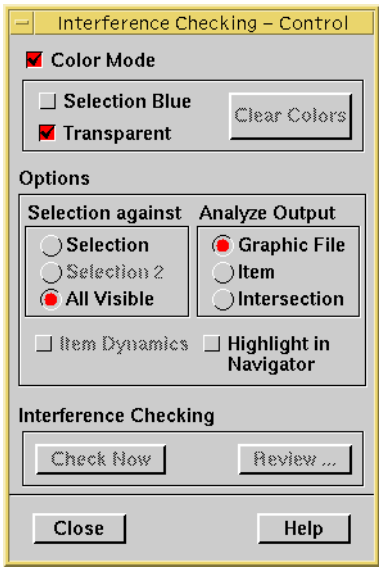
These topics are described in detail in the following sections.

# Starting Interference Checking

To start your interference check, first set up your session according to the guidelines described in the next section, then select the Check Now option. During the detection process, a beep indicates that interferences have been detected. When the checking is completed, the components that were checked are displayed in red or green to indicate the detection results. Also, the number of interferences is displayed in the status window.

# Setting Up an Interference Checking Session

To set up your interference checking session, select Interference > Control from the EPD.Visualizer window. The Interference Checking Control window appears.




The Interference Checking Control window allows you to define the interference check you want performed. This includes such actions as applying colors to the components involved in the interference check, specifying the components you want checked against the current selection, choosing how you want interferences analyzed and displayed, and reviewing and reporting the results.

## Turning On or Off Item Dynamics

You can turn interference checking on or off for item editing or item traversal operations. To do so, select the Item Dynamics option or select Interference > Item Dynamics from the EPD.Visualizer window. Interference checking can be turned on or off from the Object Path Traversal dialog box by switching the Check Collisions button in the Play area.

# Coloring and Highlighting Colliding Components

You can view colliding components in color in EPD.Visualizer as well as highlight them in EPD.Connect. Alternately, you can choose to see the bounding boxes or the edges of the colliding components. Refer to “Viewing Results Without Color” on page 5-5.

## Applying Color to Components

By default, color mode is preselected in the Interference Checking Control window, causing colors to be applied to colliding components. The colors have specific meanings:

| Color | Description |
|---|---|
| red | Indicates colliding components |
| green | Indicates noncolliding components |
| yellow | Indicates a secondary selection (Selection Set 2) |

These colors appear the first time you perform or review an interference check. The colors are applied to the wire frame, hidden line, flat shading, and smooth shading rendering methods. A secondary selection appears in yellow if that selection contains anything. You can restore the component’s original colors using the Clear Colors button or by selecting Interference > Clear from the EPD.Visualizer window.

## Viewing Results Without Color

You can choose not to apply color to colliding components by disabling color. You may prefer to see the bounding boxes or edges of the colliding components. To use boxes and edges to differentiate components, select Options > Preferences in the EPD.Visualizer window and select either boxes or edges as the highlight mode.

## Displaying the Selected Object in Blue

In addition to using red, green, and yellow to distinguish components, you can display the current selection in blue. This distinguishes the current selection from the surrounding components. This option is enabled by default in the Interference Checking Control window.

## Making Selected Objects Transparent

To make the selected parts transparent, select Selected Transparency.

### Highlighting Colliding Components in EPD.Connect

You can highlight the colliding components in the product structure tree by selecting the Highlight in Navigator option in the Interference Checking Control window. Components are highlighted for static and dynamic interference checking, except for item traversal operations. Highlighting is removed from noncolliding components that were previously highlighted in the product structure tree.

## Checking Against the Selection

During interference checking, three kinds of checking can be performed against the current selection, as follows:

- All other components in the scene are checked against the selection
- A secondary selection is checked against the selection
- All visible components in the scene are checked against the selection

With color mode enabled, you can check all components in the scene against the selection. For static interference checking, you can check the selection against a secondary selection or check the selection against itself. With color mode disabled, you can check all visible components against the selection.

### Checking Options

The following options allow you to specify the kind of check you want applied to the current selection:

- Selection

  This checks all other components in the scene against the selection, with no component in the selection being checked against itself. For example, if the selection is the contents of a graphic file, components belonging to the graphic file are not checked against each other.

Please note: With this type of selection, the red color of the colliding components may run together causing some difficulty in viewing the results.

- Selection 2

  This checks all components in the secondary selection (except those belonging to the selection) against the selection. This option is not available if the secondary selection is empty.

- All Visible

  This checks all visible components (except those belonging to the selection) against the selection.

## Creating Secondary Selections

A secondary selection gives you flexibility in the way you apply your current selection during an interference check. You create a secondary selection by copying the current selection to it. To do this, select Select > Copy Sel. -> Sel. 2 from the EPD.Visualizer window. This copies the current selection to the secondary selection, automatically clearing any previously copied secondary set. The secondary selection remains visible and colored in yellow as long as it contains at least one component. To clear the secondary selection, select Select > Clear Selection 2.

## Setting Analysis Options

During interference checking, a colliding component appears in red. This procedure is automatically enabled. However, you can specify alternative ways to analyze and display colliding components. The options available to you are the following:

- Graphic File

  When a collision is detected, the entire graphic file containing the colliding component is flagged as colliding and is displayed in red. Following the detection, any component belonging to the graphic file, including the colliding component, are ignored until the end of the detection process. Each colliding graphic file is recorded separately for reviewing.

- Item

  When a collision is detected, the item (surface containing the colliding component) is displayed in red. The colliding component itself is still used until the end of the collision detection process. Each item is recorded separately for reviewing.

- Intersection

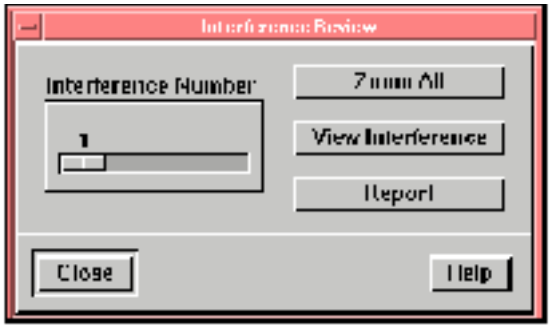
When a collision is detected between a component and an object of the selection, the intersection is displayed in red. Following the detection, the item containing the colliding component, and the component itself, are used until the end of the collision detection process. Only items and the intersection lines (not the colliding components) are recorded for reviewing.

# Reviewing the Results

You can view the results of an interference or obtain a report, as described in the following sections.

## Viewing the Interferences

To review the results of your last interference check, select the Review option. The Interference Review window appears.

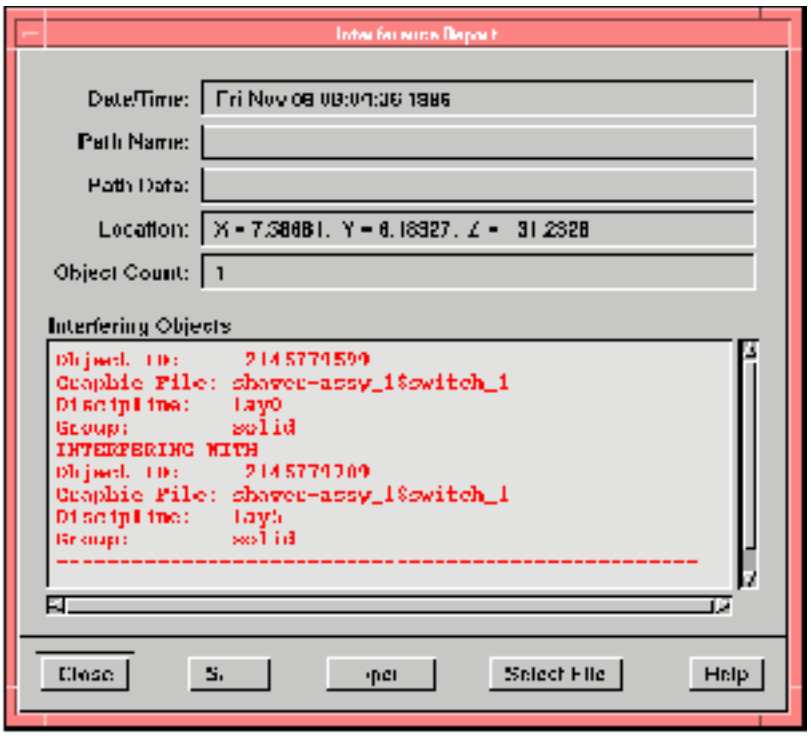



The interference number slider scale, which is used for reviewing item traversal operations, allows you to examine each operation separately. The slider scale ranges from 1 through the number of interferences, beginning with the first interference. To review an interference, move the slider to set the value for the interference number.

To enlarge your view of an interference, select View Interference. Use the palette buttons and tools to view the interference from any angle. After zooming on an interference, use Zoom All to redisplay the entire scene in its original size.

## Obtaining a Report

To obtain a report of the interference, select the Report option from the Interference Review window. The Interference Report window appears.



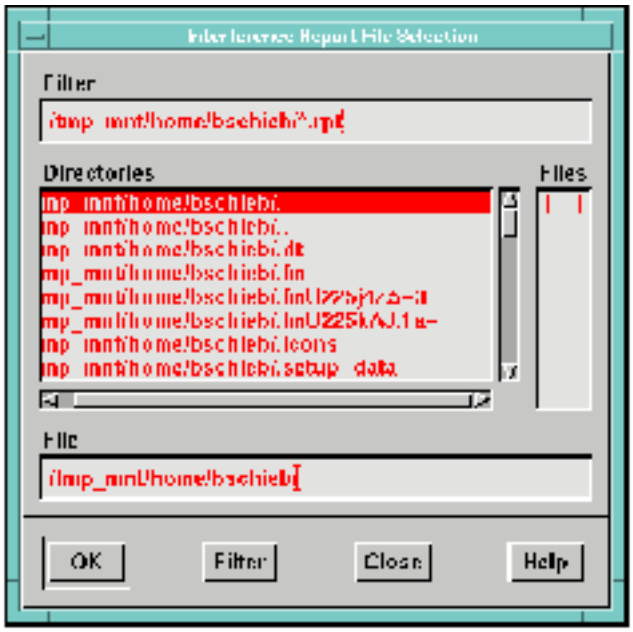

## Report Content

The information provided in the interference report is described in the following table:

| Report Item | Description |
|---|---|
| Date/Time | The date and time the interference was encountered. |
| Path Name | Name of the path the item was traversing when the interference was encountered. This information is displayed when the interference occurs during item traversal. |
| Path Data | Segment number of the path and the step number. This information is displayed when the interference occurs during item traversal. If the interference occurred during a rotation, the vertex number and rotation number are displayed. |
| Location | The XYZ coordinates of the interference location, if the interference occurred along a path. |
| Object Count | Number of interfering objects. |
| Interfering Objects | Interfering objects listed by object ID, graphic file name, discipline, and group. |

## Filing the Report

To save the interference report in a new or existing ASCII file, click Select File. The Interference Report File Selection dialog box appears. This typical dialog box is described in the section, “File Selection Dialog Box” on page 1-17.




To create a new file, enter the directory path and/or new file name in the File text box. Use the `.rpt` extension following the file name to designate your file a report file.

Click OK to return to the Interference Report window. From the Interference Report window, click Save to write the report to the file you selected or named. Click Append to append the report to the end of the file you selected. Any additional saving or appending actions are applied to the current file until you click Select File to select or name another file.

Chapter 6

# Sectioning

This chapter explains how you begin your sectioning session and includes information on the following:

- About Sectioning
- Starting Your Sectioning Session
- Displaying the Main Sectioning Window
- Defining a Plane

# About Sectioning

The sectioning capability allows you to interactively scan a complex assembly by defining and moving a plane. You can use the plane to cut away a part of the scene or to generate a section of the visible objects. Once you have enabled sectioning, the scene refreshes every time you move an object or plane.

# Starting Your Sectioning Session

To begin your sectioning session, select Sectioning from the EPD.Visualizer window. The Sectioning menu appears, providing options that allow you to do the following:

- Enable plane dynamics
- Control the plane's display and behavior

To deactivate sectioning, select Sectioning > Clear. All windows and dialog boxes related to sectioning are closed.

For a description of plane dynamics, refer to Chapter 8, "Moving a Plane."

# Displaying the Main Sectioning Window

You can perform all your sectioning activities, including plane dynamics, from one main sectioning window. To display this window, select Control from the Sectioning menu. The Sectioning Control window appears.

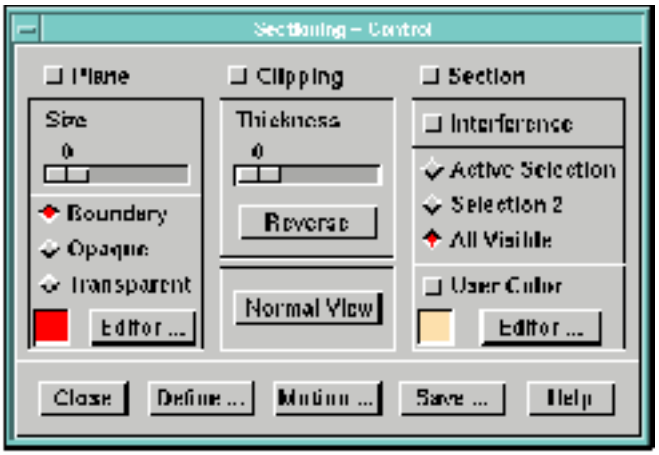


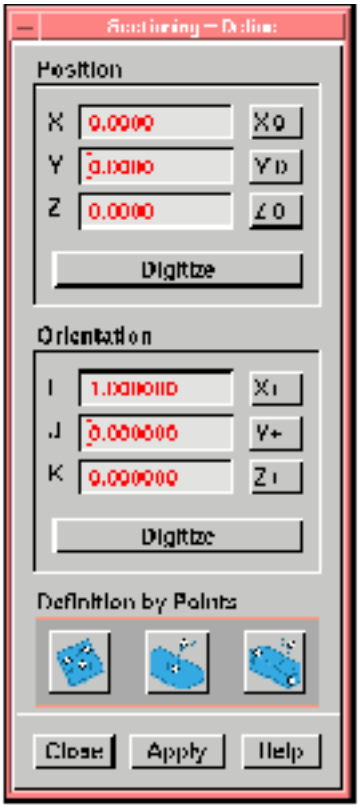

Table 6-1 provides a brief explanation of the operations you can perform from the Sectioning Control window, with pointers to sections in this chapter and to subsequent chapters where the operations are explained in detail.

**Table 6-1 Clipping and Sectioning Operations**

| Operation | For more information, refer to: |
|---|---|
| Defining a plane | “Defining a Plane” on page 6-5 |
| Displaying the plane | Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning” |
| Clipping the scene | Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning” |
| Sectioning the scene | Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning” |
| Measuring distance in sections | Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning” |
| Saving a section | Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning” |
| Setting plane dynamics | Chapter 8, “Moving a Plane” |
| Moving (translating and rotating) the plane | Chapter 8, “Moving a Plane” |

# Defining a Plane

To define the position and orientation of a plane, select Sectioning > Define from the EPD.Visualizer window. The Sectioning Define window appears.




## Positioning the Plane

The three numeric fields (X, Y, Z) indicate the center position of the plane. These are automatically updated with every movement of the plane. You can modify any value by typing a new one or by using the Digitize button. When you click the Apply button, the plane moves up to contain the new point.

If the plane has to fit the scene's extents, the center position is computed and displayed. Otherwise, the position you provide is used as the new planes center. You can use three buttons (X0, Y0, Z0) to set the corresponding value to 0.

## Orienting the Plane

The three numeric fields (I, J, K) permanently indicate the direction of the normal vector of the plane. You can modify any value by typing a new one or by digitizing a new vector. When you click the Apply button, the normal vector is unified and the plane's orientation corresponds with the new values. Three buttons (X+, Y+, Z+) give you direct access to the X, Y, or Z axis.

## Defining by Points

Three buttons with icons allow you to define a plane by digitizing vertices in the scene. The X, Y, and Z values are updated after you select the first vertex. The I, J, and K values are updated after you select the last vertex. The plane is updated when you click the Apply button.

The three iconized buttons are described in the following table:

| Use this Button: | To perform this action ... |
|---|---|
| | Define a plane by digitizing three vertices. The first vertice is the new origin of the plane. The two other vertices are contained in the plane. The X, Y, Z, I, J, and K values are updated after you select the third position. |
| | Define a plane by its origin and a point on the normal vector. |
| | Define a plane by its origin, a point in the plane, and a third point to define the normal vector. The normal vector is perpendicular to the line extending from point 1 to point 2, and passing through point 3. |

Chapter 7

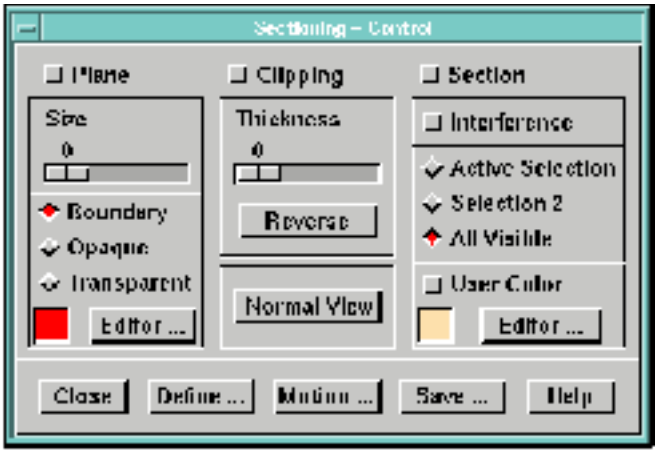
# Controlling a Plane's Display and Multiple Sectioning

This chapter describes many of the operations you can perform to scan an assembly and includes information on the following:

- Getting Started
- Displaying the Plane
- Clipping the Scene
- Sectioning the Scene
- Sectioning the Scene from Multiple References
- Measuring Distance in Sections
- Verifying a Point
- Saving a Section
- Section Window Graphics Output

# Getting Started

To specify the display and behavior of the plane, select Sectioning > Control from the EPD.EPD.Visualizer window. The Sectioning Control window appears.



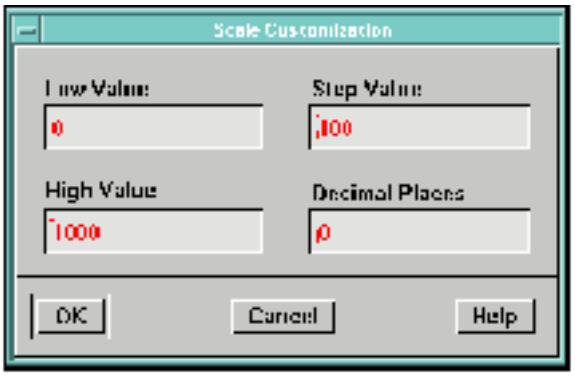
# Displaying the Plane

To display the plane used for sectioning, select the Plane option. You can display the plane as a simple contour or as a shaded or transparent surface. The normal vector to the plane appears as a short line starting from the plane's center. The center of the plane is indicated by a small, filled-in circle. The local X and Y axes of the plane are displayed as two lines belonging to the plane. The local X axis is longer than the local Y axis to aid in its identification. A plane's local axes are always used to position it at a reference. Gpath references include orientation specifications for local axes that position the plane. Enabling or disabling the plane display immediately interrupts any display update currently in process.

Unlike the movement of a plane to a reference, either global axes or the plane's local axes control the plane's rotation, depending on the option that you choose. "Getting Started" on page 8-2 describes how to rotate a plane with the Sectioning - Motion dialog box.

## Specifying the Size

The size of the plane can automatically fit the extents of the scene or you can define it. The size scale allows you to specify the size used for displaying the plane. A null value indicates that the plane has to fit the extents of the scene. Both visible and invisible objects are used to evaluate the extents. You can modify the low value, high value, step value, and the decimal places using the Scale Customization window. Click the right mouse button on the current scale value to display the Scale Customization window.



## Setting the Display Characteristics

You can specify the plane's display characteristics using the options described in the following table:

| Option | Description |
|---|---|
| Boundary | Displays the plane as a simple contour on top of the scene. With this option enabled (and with clipping and sectioning disabled), you can move the plane without refreshing the display, thus avoiding unwanted redisplays of the screen. |
| Opaque | Displays the plane as a rectangular opaque surface on top of the scene, without refreshing the entire scene. The appearance of the objects returns to normal after the next redisplay. With this option enabled, any change causes the scene or a portion of the scene to be refreshed. |
| Transparent | Displays the plane as a transparent (coefficient=0.6) rectangular surface. With this option enabled, any change causes the scene or a portion of the scene to be refreshed. |

## Defining the Color

You can use the color editor to define the color when you have set the display characteristics to opaque or transparent. When you use the color editor, the new color appears only if there is a change to the plane's display. For information about the color editor, refer to "Color Editor" on page 1-19.

# Clipping the Scene

You perform the major part of the clipping by displaying one or two cutting planes to remove the unwanted parts of the scene. By using two cutting planes with opposite norm vectors, you can display only a slice of the scene.

To clip the scene, select Clipping from the Sectioning Control window. When you enable clipping, all the geometry in front of the plane vanishes. The part of the scene behind the normal vector is retained. With clipping enabled, the scene is refreshed with any change to the plane, the clipping mode, the clipping parameters, or the section. Also, enabling or disabling clipping immediately interrupts any display update currently in process.

## Specifying the Thickness

By defining a clipping thickness, you can display all the objects behind the scene or only a slice of the scene. The thickness scale allows you to specify the clipping thickness. A null value disables display of a slice. Any other value indicates the thickness of the visible slice. You can click the right mouse button on the current scale value to display the Scale Customization window. The Scale Customization window is shown on page 7-3.

## Reversing the Normal Vector

You can use the Reverse button to invert the normal vector of the plane. If you enable reverse, the clipping is regenerated.

## Viewing the Plane

You can use the Normal View button to position the camera so it faces the cutting or sectioning plane. If a visible or hidden section exists, EPD.Visualizer adjusts the zooming factor to fit the extents of the section. If there is no section, there are two possible results as follows:

- The cutting or sectioning plane intersects the bounding box of the objects in the scene. The zooming factor fits the virtual intersection of the bounding box and the plane.
- The cutting or sectioning plane has no common part with the bounding box of the scene objects. The zooming factor fits the entire scene.

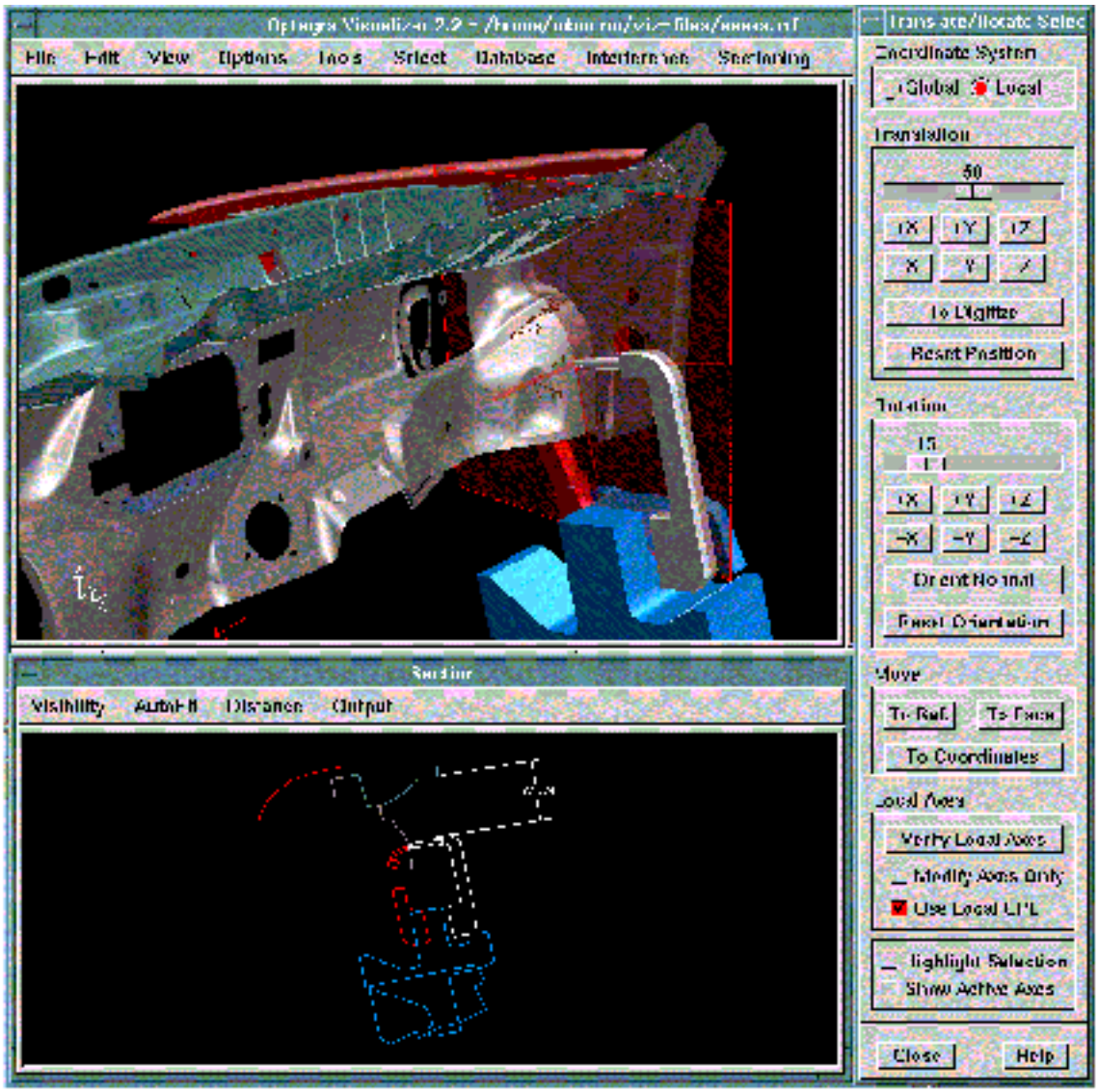
# Sectioning the Scene

The scene is comprised of graphic files, or objects, that you can section with the plane. To section the scene, select Section from the Sectioning Control window. When you enable sectioning, sectioning of the 3D objects by the plane is computed and displayed. The section is recomputed and redisplayed with every movement of the plane or any sectioned object. If the last computed section was exactly evaluated in the same plane, you can redisplay the section without updating it. However, if you do this, such changes as new colors and moved or deleted objects are ignored. Any display of the section refreshes the scene or partially refreshes it. Enabling or disabling sectioning immediately interrupts any display update currently in process. The sectioning process itself is not interrupted.

## Displaying the Section in a Separate View

When you enable sectioning, a separate Section window automatically appears. You can use this window to display the section apart from the other components in the graphics window. This increases the readability of the graphics window and

reduces the number of times it is refreshed. The Section window follows, together with windows that you might use at the same time:




You can display the section both in the graphics window and in the Section window. If the section is visible in graphics window, the scene is entirely or partially redisplayed every time you change the section. If the section is displayed in the Section window, EPD.Visualizer does not refresh the scene when clipping is disabled and the plane is displayed in Boundary mode, or when the plane is not displayed at all.

The colors of the sectioned objects are the same in the Section window as in the graphics window.

You can display the section objects colors identical to those in the graphics window or specify another color using the color editor. The User Color option in the Sectioning Control window allows you to apply the color you have selected with the color editor. Any color change performed from EPD.Connect, such as changing indicator colors or highlighting components, is immediately reflected in the section.

The Section window displays only a front view of the current section with the interference markers visible if interference checking is enabled (see "Interference Checking" on page 7-11). The window automatically adjusts to fit each newly created section. You can then scroll, zoom, or reset the view.

You can resize, move, and refresh the Section window in the same way you do other windows. The Section window automatically closes when you disable sectioning in the Sectioning Control window. You can also close it manually as you do any other window.

## Mouse Behavior in the Section Window

When working with the view displayed in the Section window, you can use the mouse to scroll, zoom, center, and reset the view, and to select objects. You use the left mouse button to select objects, and the middle and right mouse buttons to manage the view. The mouse actions that you can perform in the Section window are described in the following table:

| Desired Action | What You Need to Do |
|---|---|
| Select objects and locations on the section | Click the left mouse button. When you click on an object, EPD.Visualizer displays the name of the graphic file containing the object in the status window.<br>You can temporarily return to this default selection mode by pressing the `Shift` key while you select an object. This is necessary if you want to select an object while measuring a distance in the Section window. For detailed information about measuring distances, refer to "Measuring Distance in Sections" on page 7-20. |
| Scroll the view | Press the middle or right mouse button while moving the mouse. The view follows the cursor. |
| Zoom and unzoom | Press the middle or right mouse button while pressing the `Shift` key.<br>Move the mouse upwards to increase the zoom factor. The view remains focused on the same point. |
| Move a point to the center of the view | Place the cursor on the desired point. Click the middle mouse button without moving the mouse. (This is useful for centering a point before zooming around it.) |
| Recenter the view on the current section. | Click the right mouse button. Each time you release the button, EPD.Visualizer redisplays the view to fit the entire section. |

## View Options

Four menus are available in the Section window. These options allow you to manage the section visibility, measure distances, control the automatic view adjustment, and generate graphics output in a PostScript file. The four menus available in the Section window are described in the following table:

| Option | Description |
|---|---|
| Visibility | Displays a menu containing two options:<br>Main Window enables/disables display of the section with the other components in the graphics window. This is disabled by default.<br>References Controls the visibility of Gpath references in the sectioning plane, as explained in text following this table.<br>Squaring places a grid of lines in the Section window to assist in estimating distances.<br>Squaring Size displays selections to control the size of the grid in terms of the current unit. If the selection would create more than 100-by-100 lines, EPD.Visualizer corrects the situation by multiplying the selection by the nearest lower power of 10 that displays less than 100-by-100 lines.<br>Text Size alters the size of the text and numbers that help you interpret the scale of the Section window. You select the plus (+) or minus (-) symbol that appears following the option to make the change.<br>Squaring Center moves a grid intersection to the point that you select.<br>Squaring Recenter restores the grid intersection to its former position.<br>Clear removes the Section window and disables sectioning in the Sectioning Control window. |
| AutoFit | Fits, in the Section window, any newly created section. This is enabled by default. If this is disabled, some sections are partially or totally removed from the view, depending on the initial zooming factor. You can use the right mouse button to recenter the view on the current section. Every time you release the button, the view is redisplayed to fit the entire section. |
| Distance | Displays a menu containing five options for measuring distances between points, endpoints, and sections. For detailed information about measuring distances, refer to "Measuring Distance in Sections" on page 7-20.<br>Verify Point allows you to digitize a point as explained in "Verifying a Point" on page 7-28.<br>Drafting is explained in "Measuring Distance in Sections" on page 7-20.<br>Undo removes the last dimension, and can be selected repeatedly to remove dimensions in an order reverse to their order of origination. |
| Output | Generates PostScript output of the view shown in the Section window. The section image is generated as vectors rather than a bitmap. The position of the sectioning plane and the image scale are shown. If the two cross markers generated by distance measuring are visible when the PostScript file is generated, the markers appear in the output and the distance is indicated.<br>The Section Output dialog box is displayed when you select this option. For information about this box, refer to "Section Window Graphics Output" on page 7-36. |

When you make references visible, only the references visible in the graphics window become visible if their origin is in the sectioning plane. An origin is considered to be in the sectioning plane if its distance from the sectioning plane is less than 0.005 units. Any motion of the sectioning plane or the creation of a new Gpath updates the Section window.

References are represented in two ways in the Section window, depending on the orientation of their PVS_SECTION_AXIS axis. The PVS_SECTION_AXIS environment variable controls which axis is used as a normal in orienting the plane relative to the reference. *Normal* means 90 degrees plus or minus 0.5 degrees, providing a tolerance of one-half degree.

The default value of PVS_SECTION_AXIS is X. You can set PVS_SECTION_AXIS to any of the following values: `X`, `Y`, `Z`, `x`, `y`, or `z`. If a reference has a PVS_SECTION_AXIS axis normal to the plane, the reference has a full representation in the Section window, consisting of a name and two axes. If a reference has a PVS_SECTION_AXIS axis that is not normal to the plane, the reference has a partial representation in the Section window, consisting of a plus sign (+) at its origin and a name.The color, text, and axes sizes of references are the same in the graphics window and the Section window.

You can get information about a reference that appears in the Section window. Clicking on the representation of the reference in the Section window displays the name of the Gpath and reference in the status window.

# Interference Checking

You can enable or disable interference checking from the Sectioning Control window. With interference checking enabled, you can highlight the common points of sections obtained by cutting different graphic files. When a section is displayed, you can enable interference checking to highlight the common points, without regenerating the section. Interferences are highlighted in red in the section. When you enable interference checking, the Sectioning - Interference dialog box appears.

In the Sectioning - Interference dialog box, you can enter values for proximity tolerance and contact distance. The number of interferences detected is displayed in the dialog box as well as in the status window of the main graphics window.

Interference checking uses the values you enter for proximity tolerance and contact distance to detect interference between components that are close together. If you enter illegal characters (such as alphabetical characters and punctuation signs) and negative values, EPD.Visualizer displays the following error message:

```
Illegal value was entered!
```

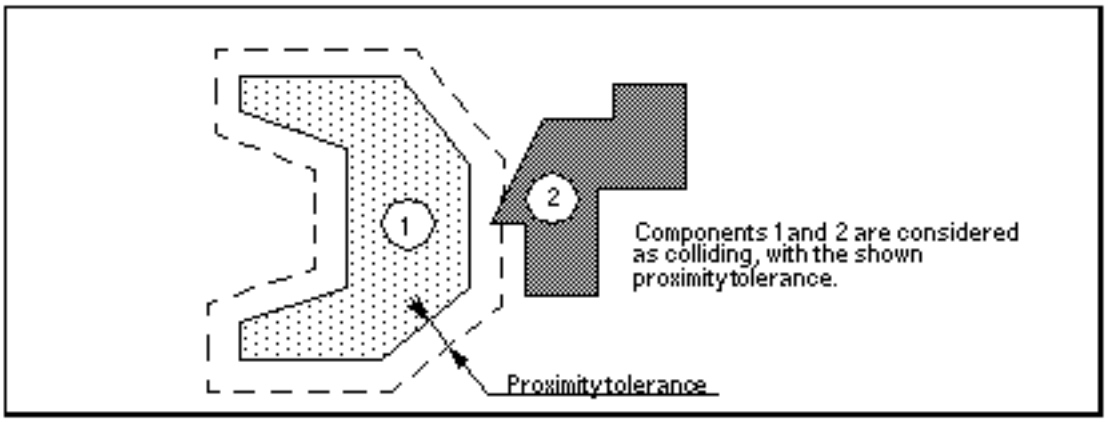
If you enter null values for proximity tolerance and contact distance, only real intersections between components are considered as interferences. When you change the values for proximity tolerance and contact distance, the previous number of interferences is removed and replaced by question marks (???). This indicates that you need to update the interference checking according to the new values. You can perform the update in one of three ways:

- By moving the sectioning plane
- By pressing `Return` in the either the Proximity tolerance or Contact distance field
- By disabling and reenabling interference checking in the Sectioning Control window

After entering values, you can close the Sectioning - Interference dialog box from its frame menu. When the dialog box is closed, you can see the number of interferences in the status window of the main graphics window. The dialog box also disappears when you disable interference checking in the Sectioning Control window.

## Setting a Proximity Tolerance

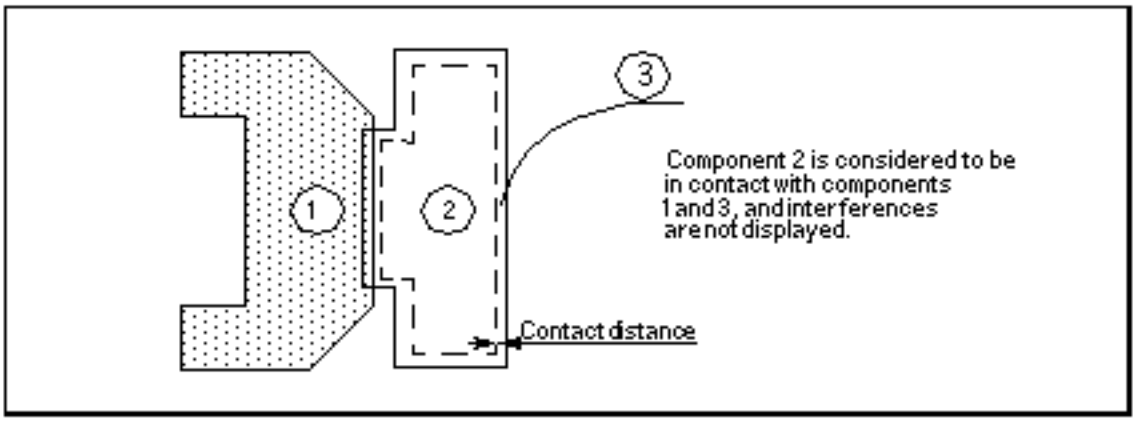
The proximity tolerance indicates the minimum distance used to determine if two components are colliding in the sectioning plane. This allows you to include sections that are close together in the interference check. By increasing the proximity tolerance, you can increase the number of potential interferences found in the section. Proximity tolerance is applied to the left and right sides of each segment belonging to the section, as shown in the following illustration:




Please note: If you enter a large value for proximity tolerance, calculations may be lengthy and the results will show almost all components as interfering.

### Setting a Contact Distance

The contact distance allows some interferences to be considered as contacts between components but not counted as interferences. By increasing the contact distance, you reduce the number of interferences detected. Interferences that are considered as contacts are ignored. As with proximity tolerance, the contact distance is applied to both sides of each segment belonging to the section, as shown in the following illustration:


3
1
2
Component 2 is considered to be in contact with components 1 and 3, and interferences are not displayed.
Contact distance

Please note: If you provide a contact distance that is greater than the proximity tolerance, the proximity tolerance is ignored.

## Selecting the Objects To Be Sectioned

You can use three options (Active Selection, Selection 2, or All Visible) to specify the objects you want sectioned. These options are described in the following table:

| Option | Description |
|---|---|
| Active Selection | Sections all objects in the current selection. |
| Selection 2 | Sections all objects in the secondary selection. This option is only available if the secondary selection contains at least one object. |
| All Visible | Sections all visible objects. |

You need to initialize the secondary selection by copying the current selection to it. To do this, select Select > Copy Sel. -> Sel. 2 from the menu bar in the EPD.Visualizer window. The contents of the secondary selection are cleared and the current selection is copied into it. As long as it contains at least one object, the secondary selection remains visible and colored (in wheat).

Please note: With sectioning enabled, the section is regenerated each time you change the secondary selection.

# Sectioning the Scene from Multiple References

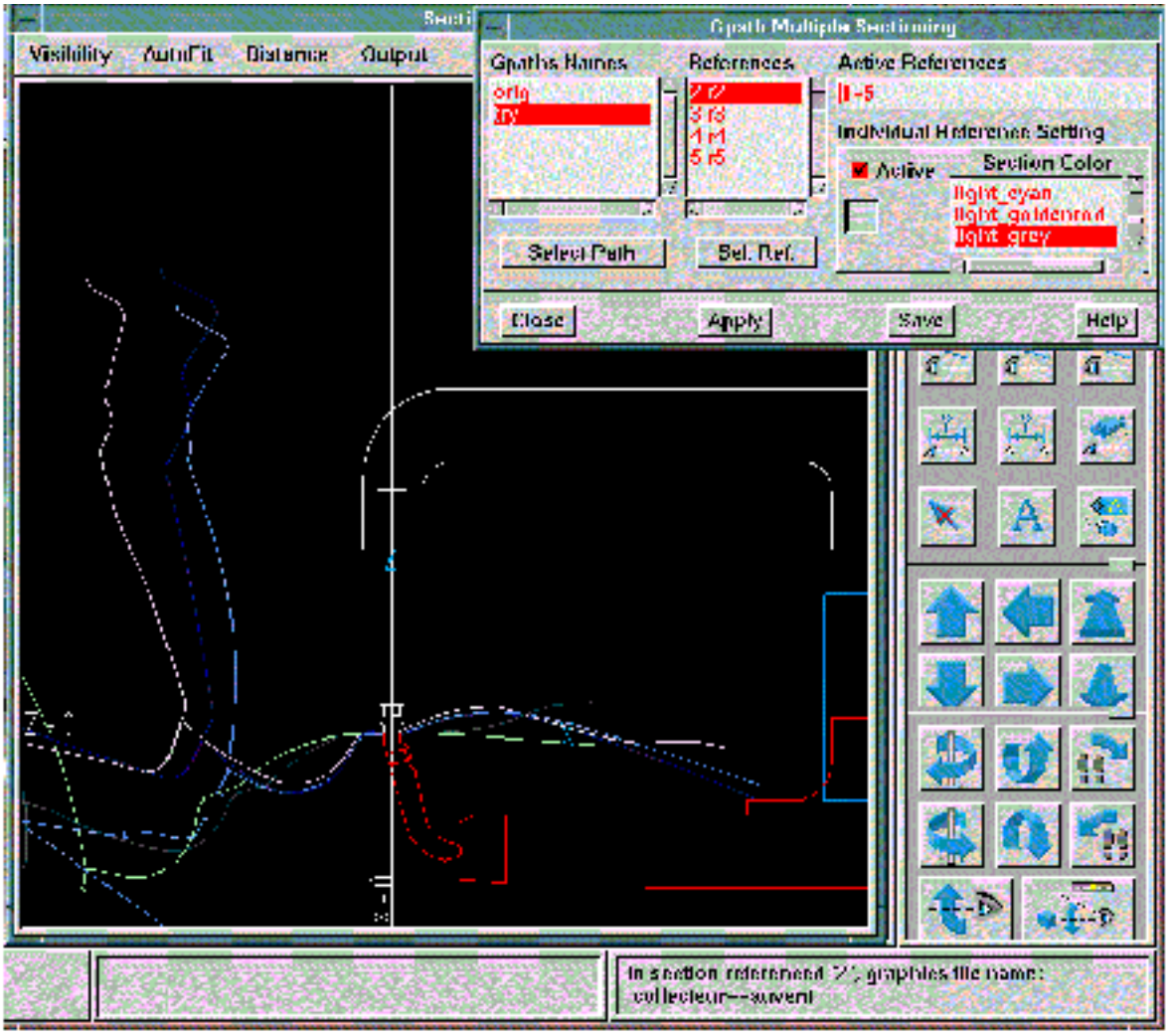
Sectioning a scene from multiple references allows you to display simultaneously multiple sections of the 3D scene created by a plane that moves to each reference. You can attach an object to the sectioning plane. Sectioning a scene from multiple references allows you to visually analyze interference between a dynamic object and static objects. Sectioning a scene from multiple references indicates how the shape of a set of surfaces changes along a profile.

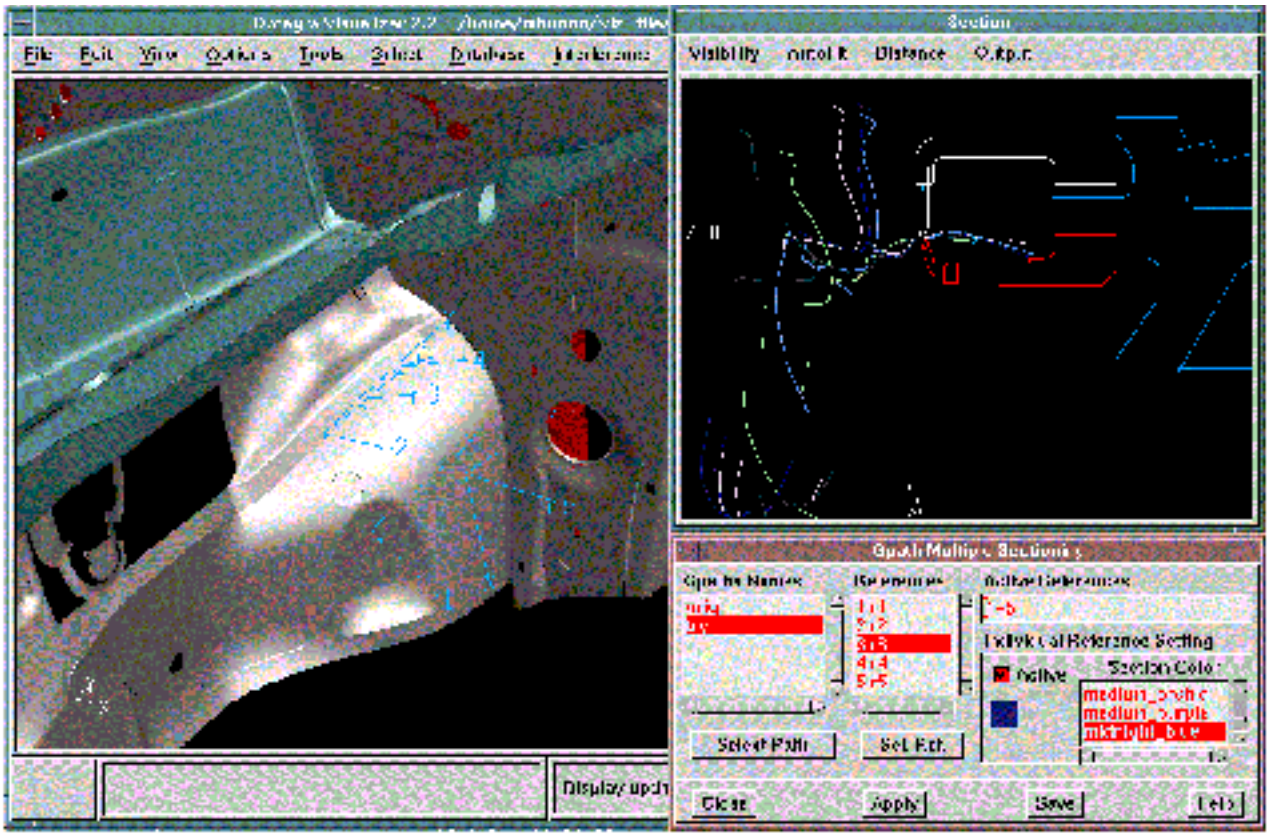
The topic "Path Editor" on page 3-28 explains in detail what Gpaths are and how to create them. A Gpath is a series of points called references, each of which is associated with an orientation. You can attach a sectioning plane to an object's referential and move the attached pair from reference to reference in a Gpath. At each reference the sectioning plane creates a new section that appears in the Section window, revealing any interference. All the sections appear superimposed in the Section window with a common origin and orientation. The dynamic object has the same representation in all the sections, so that each section represents the 3D scene as though it had moved around the object relative to other sections.

Objects should not move while you section the scene from multiple references. Moving an object means that you need to perform the sectioning again. EPD.Visualizer's other sectioning features available from the Section menu should not be used in conjunction with sectioning the scene from multiple references, and the Section menu becomes unavailable and windows opened from it close to prevent an inaccurate Section window.

You can save the results of multiple sectioning to a EPD.Visualizer textual graphic file.

The following two figures show typical results of multiple sectioning.

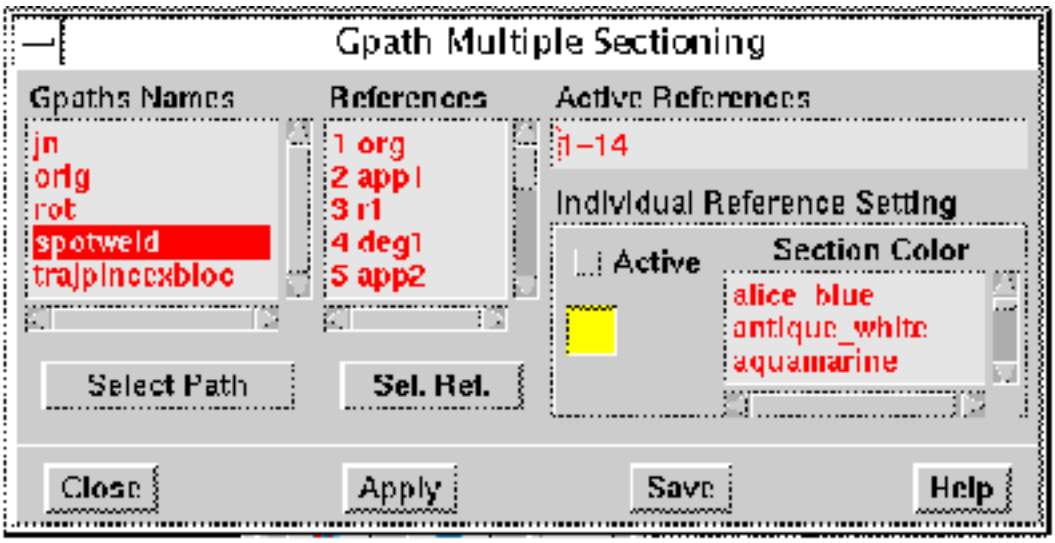



File
Edit
Tools
Select
Visibility
Distance
Select Path
Clear
Apply
Save
Active
Section Color

# Configuring Multiple Sectioning

You specify multiple sectioning in the Gpath Multiple Sectioning window. You display the window by selecting Edit > Item Dynamics > Gpath Sectioning. You can open the window in either of the following circumstances:

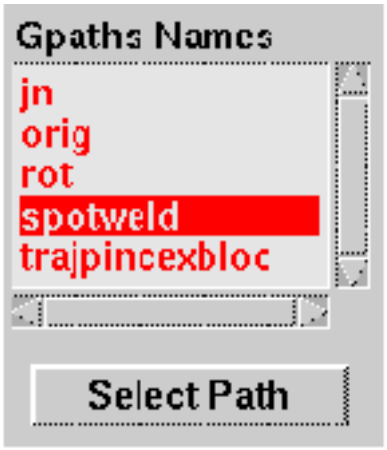
- The selection list contains at least one object.
- The sectioning plane is specified as Dynamic. Selecting Sectioning > Plane Dynamic makes this specification, which allows multiple sectioning by a plane with no attached object.



The following steps specify multiple sectioning:

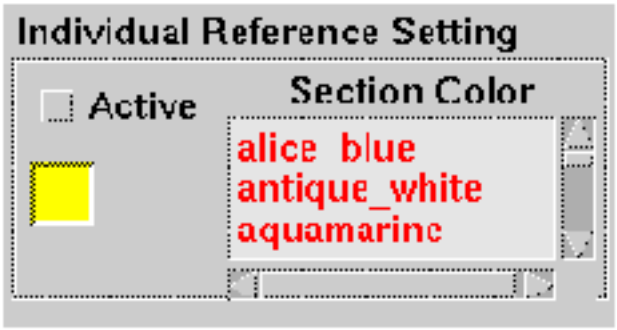
1. Select a Gpath. You can click on a path in the Gpaths Names scrolling list or press the Select Path button and select a Gpath in the graphics window. If the Gpath has not been used in a multiple sectioning process, its settings are defaults: all references are active and a different color is assigned to each section. If the Gpath has been used in a multiple sectioning process, all its settings conferred for that process are restored.

   Selecting a Gpath places all its references in the References list with a number located at the left side of each reference.




2. Modify the Active References field. Active references are those to which the dynamic object and its linked plane move. Numbers that appear on the left side of active references in the References list are represented in the Active References area.

   You can control whether a reference is active by clicking on it in the References list and selecting the Active button in the Individual Reference Setting area.

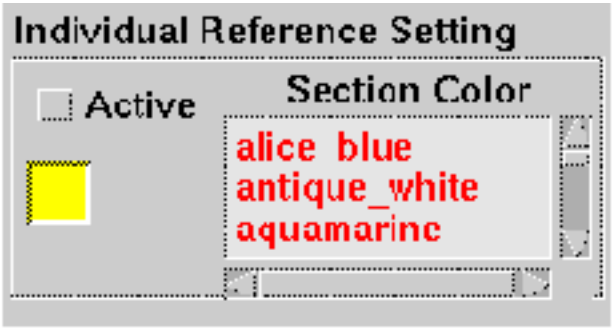


You can compose a list of active references by typing in the Active References area. Two types of entries are possible, and all entries must be separated by spaces:

- Single references — Type the number associated with the reference in the References scrolling list.
- Inclusive ranges — Type the numbers associated with the endpoint references in the References scrolling list and separate them with a minus sign (-).

The reference status in the Active References area rules if it is inconsistent with a setting created with the Active toggle button. Changing a reference's active status with the Active toggle button immediately changes the Active References list.

You set the color of a reference by clicking it in the references list and clicking selecting the desired color in the Section Color scrolling list in the Individual Reference Setting area. Colors whose names exceed 15 characters do not appear in the scrolling list.




You can select a new reference to configure and a new Gpath, if the reference is in a new Gpath, by pressing the Sel. Ref. button and selecting a reference in the graphics window.

## Performing Multiple Sectioning

After configuring the multiple sectioning, start it by pressing the Apply button in the Gpath Multiple Sectioning window and observe the results in the Section window which appears automatically. Each section appears in a specific color in the Section window. You can use the tools discussed in this chapter to analyze the sections. You can select another object and press the Apply button again.

From the Section window, you can identify a sectioned object and the reference that oriented the sectioning plane when the section holding the object was created. Click on the sectioned object in the Section window to show the following information:

- The reference is selected in the References list in the Gpath Multiple Sectioning window.
- The name of the graphics file for the sectioned object appears in the status window.

## Saving Multiple Sectioning

Pressing the Save button in the Gpath Multiple Sectioning window saves all the sections in the Section window to a `.gaf` file. The name of the file is formed by adding the suffix `.gaf` to the name of the Gpath. You can define the name of the directory in which to save the files by setting the CVDPA_SECTIONS_DIR environment variable. Appendix B, "EPD.Visualizer Textual Graphic File" explains the format in which files are saved.

# Measuring Distance in Sections

You can evaluate gaps between objects in the sectioning plane and measure distances in sections. The entities between which you can measure distance are points, endpoints, and whole sections. To measure distances, select Tools > Measure Distance from the EPD.Visualizer window menu bar or select the Distance menu from the Section window. Five tools are available to you for measuring distance in sections. The following sections discuss those tools.

The five tools have a common feature: you can display dimensions in the Section window obtained by the measuring tool that you select. Select the Drafting button in the Section window's Distance menu to activate this feature. After you measure a distance with the tool you select, a message in the prompt window requests you to indicate a location near the boundary of the Section window. The boundary that you select determines the type of dimension and the location of the dimension's baseline:

- Left or right boundary selected: vertical dimension
- Top or bottom boundary selected: horizontal dimension
- Press the Control key while selecting the objects to measure the shortest distance between them on the sectioning plane, known as "parallel dimensions."

Dimensions appear near the Section window boundary. To avoid overlaps you can scroll the window before making new dimensions. Selecting Distance > Undo removes the last dimension, and you can repeat the operation to remove additional dimensions in a sequence reverse to that in which they were made. Pressing the right-hand mouse button in the Section window removes all dimensions and recenters the Section window on the extents of the section.

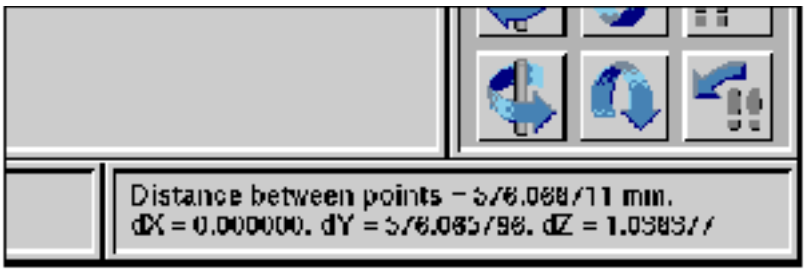
The dimensions have the following parts:

- The horizontal, vertical, or real distance in the current unit. The text size is 1/80 the size of the Section window's maximum dimension, and the size of the last inserted dimension is applied to all existing ones.
- One baseline with no arrows
- Two extension lines

## Distance Between Two Points

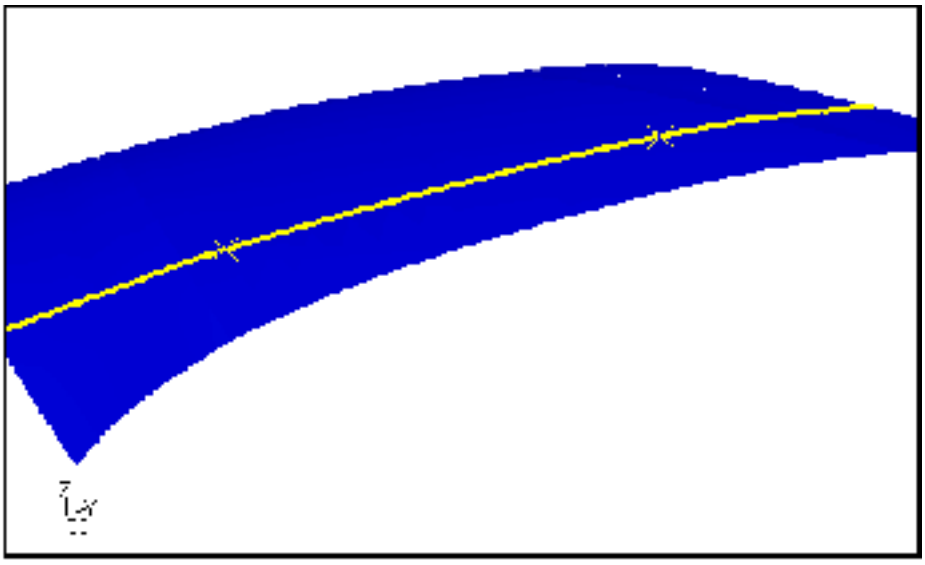
You can use Sections Points to measure the distance between any two points on a section. With this tool, you need to digitize two points. You digitize a point by clicking the left mouse button on a location in a visible section. You can digitize the points in the main view, if the section is visible in it, or in the Section window. Also, you can digitize vertices as well as points between two vertices of a segment.

Cross markers appear on the digitized points. The distance between these points, and the distances along the X, Y, and Z axes, are displayed in the status window, as follows:

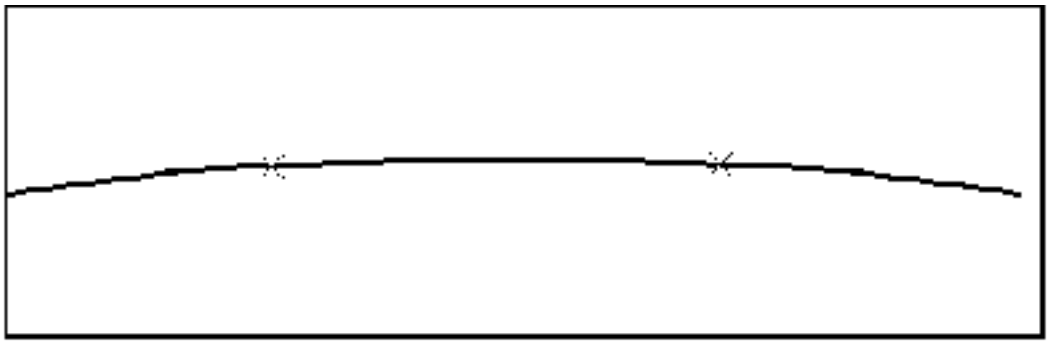



This tool remains active until you select another tool. The markers remain until you select another tool or start a new measure. This behavior applies to all the measuring tools described in this section.

The following illustration shows the main view with two digitized points on a section.



The following illustration shows the digitized section (shown in the previous illustration) as it appears in the Section window:



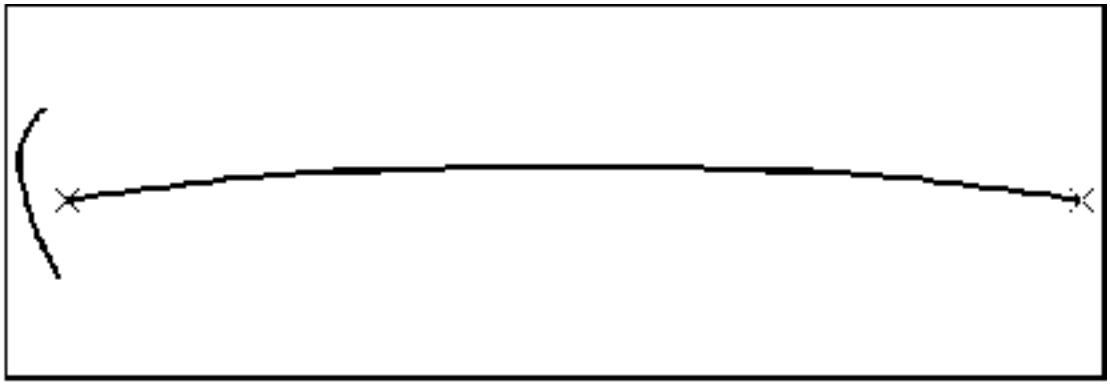
## Distance Between Two Endpoints

You can use Sections Ends to measure the distance between two endpoints of the same section or of two different sections. With this tool, you need to digitize two points in the same way you do for the other measuring tools. When you digitize two points, EPD.Visualizer selects the nearest endpoints. If there is no endpoint, which is the case if the section is a closed polygon, EPD.Visualizer accepts the digitized point instead of an endpoint.

Cross markers appear on the endpoints. The distance between these points, and the distances along the X, Y, and Z axes, are displayed in the status window. The tolerance that EPD.Visualizer uses for chaining the polylines of the sections to determine the endpoints is the gap value you enter in the Sectioning - Save dialog box. The default value is 0.1. The most appropriate tolerance is the tessellation tolerance or a tolerance a little greater than that. For more information, refer to "Considerations" on page 7-27. For information about the Sectioning - Save dialog box, refer to "Saving a Section" on page 7-29.

As with the other measuring tools, this tool remains active until you select another tool. The markers remain until you select another tool or start a new measure.

The following illustration shows two endpoints of the same section.



The following illustration shows endpoints of two different sections.

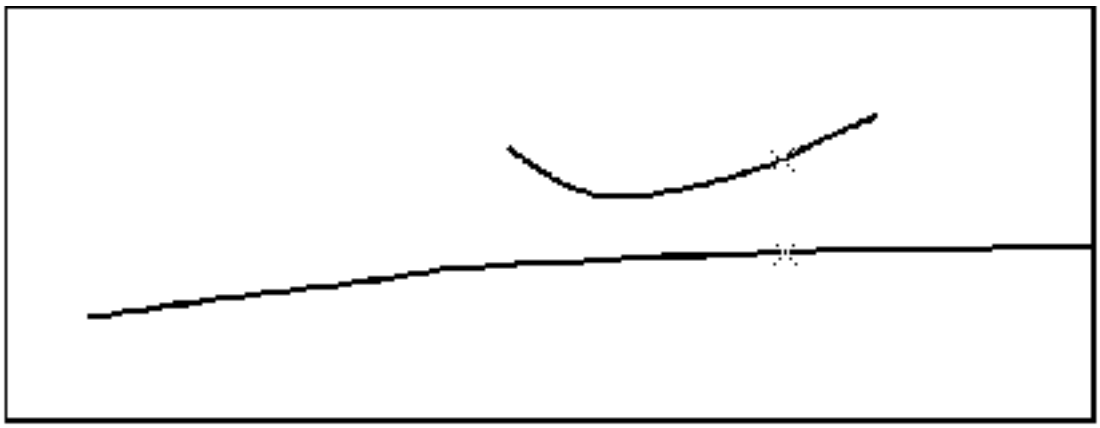
## Minimum Distance Between a Point and a Section

You can use mini Point & Section to measure the minimum distance between any point of a section, and another section, and to see where the minimum distance is reached. With this tool, you need to digitize a point and a section. EPD.Visualizer accepts the first digitized point as is. It uses the second digitized point to select the section associated with the point. For the first point, you can digitize vertices as well as points between two vertices of a segment.

If you digitize the first and second points in the same section, EPD.Visualizer displays a warning message in a small panel in the middle of the screen, and it reinitializes the tool. You have to reselect the point to use it for another measuring distance operation.

Cross markers appear on the first point, and on the point of the section point where the minimum distance occurs. The distance between these points, and the distances along the X, Y, and Z axes, are displayed in the status window.

The following illustration shows a digitized point on a section (upper) and the point of minimal distance between it and a second section (lower). The cross marker on the second section indicates the point at which the minimum distance occurs.



## Minimum Distance Between an Endpoint and a Section

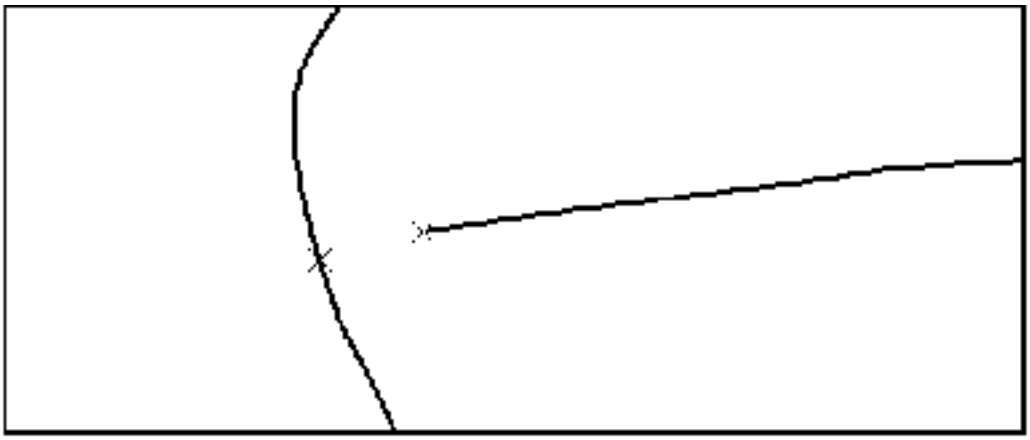
You can use mini End & Section to measure the minimum distance between an endpoint of a section, and another section, and to see where the minimum distance is reached. With this tool, you need to digitize a point and a section. When you digitize the first point, EPD.Visualizer selects the nearest endpoint. If there is no endpoint, which is the case if the section is a closed polygon, EPD.Visualizer accepts the digitized point instead of an endpoint. It uses the second digitized point to select the section associated with that point.

If you digitize the first and second points in the same section, EPD.Visualizer displays a warning message in a small panel in the middle of the screen, and it reinitializes the tool. You have to reselect the point to use it for another measuring distance operation.

Cross markers appear on the endpoint, and on the point of the section where the minimum distance occurs. The distance between these points, and the distances along the X, Y, and Z axes, are displayed in the status window.

The tolerance that EPD.Visualizer uses for chaining the polylines of the sections to determine the endpoint is the gap value you enter in the Sectioning - Save dialog box. The default value is 0.1. The most appropriate tolerance is the tessellation tolerance or a tolerance a little greater than that. For more information, refer to "Considerations" on page 7-27. For information about the Sectioning - Save dialog box, refer to "Saving a Section" on page 7-29.

The following illustration shows the digitized endpoint of a section (right) and the point of minimal distance between it and a second section (left). The cross marker on the second section indicates the point at which the minimum distance occurs.

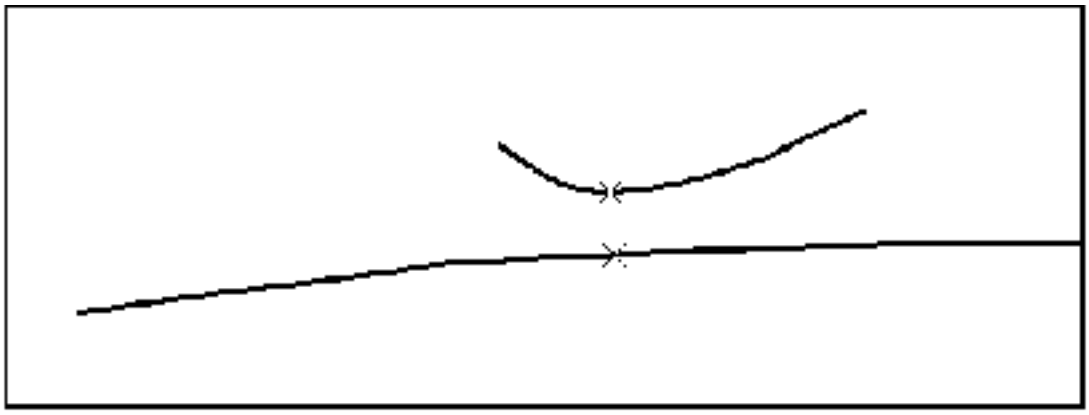


## Minimum Distance Between Two Sections

You can use mini Sections to measure the minimum distance between two sections and to see where the minimum distance is reached. If the two sections are crossing, EPD.Visualizer shows the crossing points. With this tool, you need to digitize two sections. EPD.Visualizer uses the digitized points to select the associated sections.

Cross markers appear on points where the minimum distance occurs. The distance between these points, and the distances along the X, Y, and Z axes, are displayed in the status window. If the two digitized sections are the same, EPD.Visualizer displays a warning message. Also, if the sections are crossing, EPD.Visualizer displays a warning message and cross markers to indicate the points where the sections cross.

The following illustration shows two sections. Cross markers indicate the point at which the minimum distance between them occurs.



## Considerations

Some restrictions and limitations that apply to measuring distances are the following:

- You may want to identify an object in the Section window while measuring a distance. To identify an object without disabling distance measuring, press the `Shift` key while selecting the object in the Section window with the left mouse button. For information about selecting objects in the Section window, refer to "Mouse Behavior in the Section Window" on page 7-8.
- You must enter an appropriate tolerance (gap value) for EPD.Visualizer to use for chaining the polylines of sections to determine the endpoint. Inappropriate values can lead to strange results when you digitize an endpoint. It is recommended that you use the tessellation tolerance or a tolerance a little greater than that.
- Two markers that are too close together can hamper your view and disappear in some views.

# Verifying a Point

You can digitize points in the Section window. Select Distance > Verify Point and select a point in the Section Window. An X temporarily displays at the digitized location. You see the following digitized information in the status window:

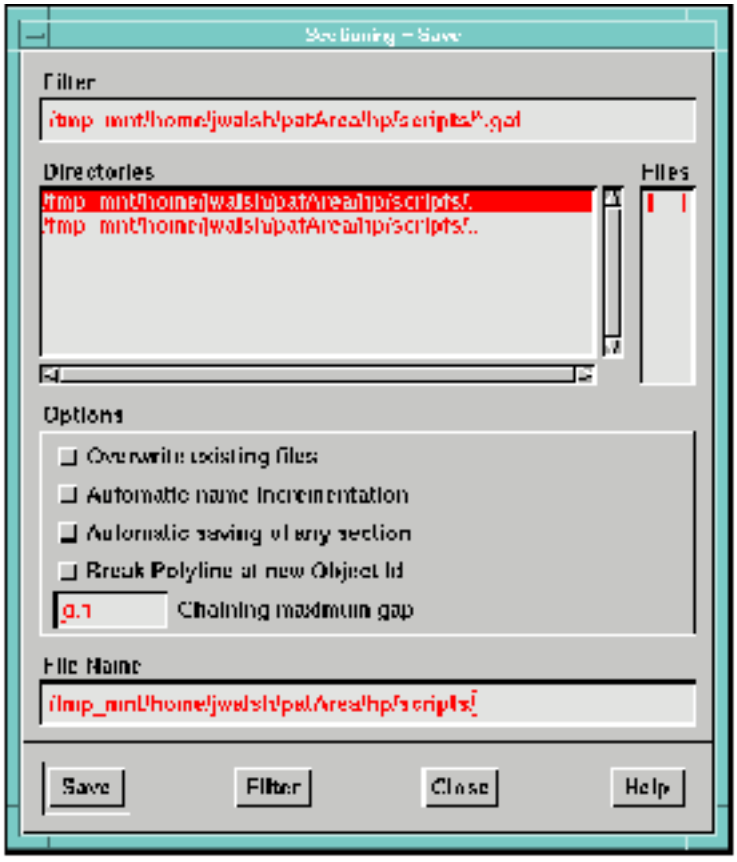
- current units
- coordinates of the selected point expressed in current units relative to global coordinates

# Saving a Section

You can save one or more sections in an external text (`.gaf`) file that EPD.Visualizer generates automatically for each motion of the cutting plane, or that you generate manually. When you save a section, it appears as a collection of objects. There is one object for each sectioned graphic file in the scene and one object for each position of the cutting plane. Each object is comprised of one or more polylines.

A single graphic file may contain a complex assembly. EPD.Visualizer uses the attributes (`group` and `object_id`) to indicate the position of the cutting plane and the name of the sectioned graphic file. Interferences between graphic files, when detected by the sectioning process, are stored in the `.gaf` file as comments.

You use the Save button in the Sectioning Control window to save sections in a `.gaf` file. A Sectioning - Save dialog box appears.



In the Sectioning - Save dialog box, you can use the Filter, Directories, Files, and File Name fields to select the output file. Generally, these fields behave in the same way as similar fields in other dialog boxes. Refer to "File Selection Dialog Box" on page 1-17 for more information. However, some rules apply specifically to saving sections, as follows:

- The initial values of the Filter and File Name fields are defined by the CVDPA_SECTIONS_DIR environment variable. if this variable is not defined, EPD.Visualizer uses the current directory.
- If you change the directory displayed in the Filter or File Name fields, EPD.Visualizer ignores the directory defined by CVDPA_SECTIONS_DIR until you select File > New Session.
- You can provide the name of a new directory or file by defining it with CVDPA_SECTIONS_DIR or by typing it. EPD.Visualizer creates all the necessary subdirectories when saving a section in the new directory or file.
- When typing the names of `.gaf` files, you can omit the `.gaf` suffix. EPD.Visualizer automatically appends the suffix if it is missing.

## Save Options

Options allow you to save one or more sections in a `.gaf` file. Each newly created section can be automatically added to the current `.gaf` file or saved in a new file. You can define the file names or have EPD.Visualizer generate them by adding a number to the name you define.

Five options are available in the Sectioning - Save dialog box. Three options assist you in managing the creation of the output files. Two of the options define the criteria EPD.Visualizer uses to generate polylines for the section. The options are described in the following table:

| Option | Description |
|---|---|
| Overwrite existing files | With this option enabled, you allow EPD.Visualizer to overwrite an existing file without your permission. This may be useful with the Automatic saving of any section option enabled. By default, when saving a section, you have to confirm file overwriting by EPD.Visualizer. |
| Automatic name incrementation | With this option enabled, EPD.Visualizer increments the file name by a step of 1 when you save a section. It saves only one section in a file and automatically increments the file name for each section. If the file name does not end with a digit, EPD.Visualizer automatically adds a zero (0) when saving the first section. By default, EPD.Visualizer uses the file name that you provide, with no incrementation. |

| Option | Description |
|---|---|
| Automatic saving of any section | With this option enabled, EPD.Visualizer automatically saves any newly created section. However, this option interacts with the Automatic name incrementation option in the following ways:<br>With Automatic name incrementation is enabled, each new section is saved in a new file, with only one section per file.<br>With Automatic name incrementation is disabled, while the file is open, each new section is added to the current file. The file is automatically closed when you perform any of the following actions:<br>start a new session<br>select Sectioning > Clear<br>select a new file name<br>disable Automatic saving of any section<br>enable Automatic name incrementation<br>You can close the file or leave it open when you perform the following actions:<br>disable sectioning<br>rotate the plane (See"Rotating the Plane" on page 8-8.)<br>define a new plane (See "Defining a Plane" on page 6-5.)<br>EPD.Visualizer indicates that the sectioning parameters have changed and asks whether or not you want to close the file. |
| Break Polyline at new Object Id | With this option enabled, several objects belonging to the same graphic file will produce several polylines in the `.gaf` file. Each polyline will belong to a single object of the sectioned graphic file. By default, saving a section may generate a single polyline for several contiguous objects belonging to the same graphic file. |
| Chaining maximum gap | EPD.Visualizer uses the gap value to merge the segments of the section into continuous polylines. If there is a hole greater than the gap, a new polyline is opened. |

Click the Save button to manually generate the `.gaf` file. The section, if one exists, is written to the file according to the options you select.

# Output File Format

The text (`.gaf`) file that is generated when you save a section has the following format:

```
#GRAPHICS_FILE
 #CONTROL_OPTIONS
  unit = .....
 #ENDmenu
```

The unit is defined from the Options > Preferences (Working Unit)

```
#EXTENTS
 xmin = .....
 ymin = .....
 zmin = .....
 xmax = .....
 ymax = .....
 zmax = .....
#END
```

The extents are not provided if several sections are saved in the same graphic file

```
< Desc. of section 1 >
!
< Desc. of section 2 >
!
..../....
< Desc. of section n >
#END
```

## Description of a Section

A section description (shown as `Desc. of section` n in previous example) follows:

```
< Desc. of section of graphic file 1 >
< Desc. of section of graphic file 2 >
..../....
```

Each graphic file intersected by the cutting plane generates one object

```
< Desc. of section of graphic file n>
< List of interferences >
```

## Description of a Sectioned Graphic File

Each sectioned graphic file produces a portion of the `.gaf` file. A description of a sectioned graphic file (shown as `< Desc. of section of graphic file 1, file 2, ... file n >` in the previous example) follows:

```
#COLORS
 yellow = (1.000000 1.000000 0.000000)
#END
#OBJ
 object_id = name_graphic_file
 discipline = section
 group = "a-0.89463_b-0.34467_c-0.28433_d2343.4099"
 color = yellow
 #PL
  43.850868 67.349365 -159.731003
  43.256786 68.456612 -159.924911
  42.777077 69.621246 -159.749084
  42.452496 70.735733 -159.228958
  42.309612 71.702507 -158.414810
 #END
 #PL
  2765.166504 -838.056274 557.356689
  2759.955566 -837.041016 572.521973
  2758.226074 -836.646484 577.484985
 #END
#END
```

An explanation of this example follows:

- The color is that of the first object found in the graphic file. With EPD.Visualizer, associating a single color to a graphic file is common practice.
- The attribute `object_id` contains the last thirty-one characters of the name of the sectioned graphic file. The suffix (`.gaf` or `.gbf`) is omitted.
- Always set the attribute `discipline` to `section`. You can then use Select > Build Selection List to incorporate in the current selection all objects having the `discipline` attribute equal to `section`. This allows you to select and manipulate all the sections in a single operation.
- The attribute `group` indicates the coefficients defining the cutting plane ($aX + bY + cZ + D = 0$).

## Listing Interferences

The list of interferences is generated only if interferences are detected. Because there is no keyword allowing you to describe a point in the `.gaf` format, interferences appear as comments in the `.gaf` file. A comment begins with an exclamation mark. The first comment indicates the number of interferences found and the coefficients defining the cutting plane. The last comment closes the list of interferences. An example of comments follows:

```
 ! NB_INTERFERENCES 2 in A=-0.89463 B=-0.34467 C=-0.28433
D=2343.4099
    !  INTERFERENCE X2765.086914 Y-837.465027 Z556.890625
    !  INTERFERENCE X2755.194580 Y-834.280701 Z584.155640
    ! END_INTERFERENCES
```

## Example File

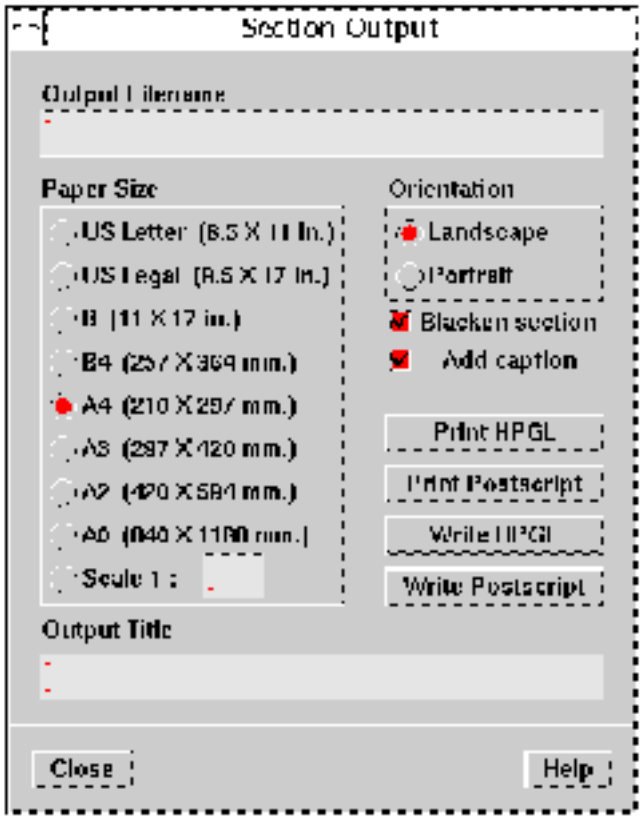
An example output file follows (it consists of one cutting plane, two graphic files, and two interferences):

```
#GRAPHICS_FILE
 #CONTROL_OPTIONS
  unit = millimeters
 #END
 #EXTENT
  xmin = 2457.839600
  ymin = -838.364746
  zmin = 548.950195
  xmax = 2767.957031
  ymax = -541.407837
  zmax = 1164.738159
 #END
 #COLORS
  red = (1.000000 0.000000 0.000000)
 #END
 #OBJ
  object_id = "aile-ar-g_1"
  discipline = section
  group = "a-0.89463_b-0.34467_c-0.28433_d2343.4099"
  color = red
 #PL
  2457.839600 -541.407837 1164.738159
  2462.712891 -549.393127 1159.084351
  2551.934814 -660.151184 1012.616943
  2633.821289 -735.565247 846.384338
  2636.415283 -738.466187 841.739136
  2767.957031 -838.364746 548.950195
 #END
#END
#OBJ
 object_id = "bas-aile-ar-g_1"
 discipline = section
```

```
        group = "a-0.89463_b-0.34467_c-0.28433_d2343.4099"
        color = red
        #PL
         2740.700195 -823.388245 616.557495
         2751.208008 -831.201294 592.966553
         2758.264648 -836.652039 577.370728
         2765.140137 -835.266846 554.058105
         2763.268555 -829.398499 552.833191
         2762.095703 -825.214661 551.451355
         2761.740479 -824.028198 551.130737
        #END
       #END
! NB_INTERFERENCES 2 in A=-0.89463 B=-0.34467 C=-0.28433
D=2343.4099
   !  INTERFERENCE X2765.086914 Y-837.465027 Z556.890625
   !  INTERFERENCE X2755.194580 Y-834.280701 Z584.155640
   ! END_INTERFERENCES
   #END
```

# Section Window Graphics Output

You can write and print graphics developed from the Section window by making selections in the Section Output window. You display the Section Output window by selecting Output > Section Output in the Section window.




To write the contents of the Section window as a file you perform the following actions:

1. Set the options:
   - Paper Size — If one of the standard paper formats is selected, the contents of the Section window will be automatically scaled to fit the paper size and to adjust for the orientation. Selecting Scale 1: x and entering a scale value in the adjacent field adjusts the size of the graphic that prints. The choice of an inappropriate scale may cause some graphics to disappear.
   - Orientation — Select Landscape or Portrait.
   - Blacken Section — Selecting this makes the section print black. Clearing it specifies the colors in the Section window. The names of sectioned objects are the same colors as the sections.

- Add Caption — Selecting this makes a list of all sectioned objects, in which the names appear in the same colors as the objects.
- Output Title — Typing a title in this field makes that title appear as a comment in the printout.

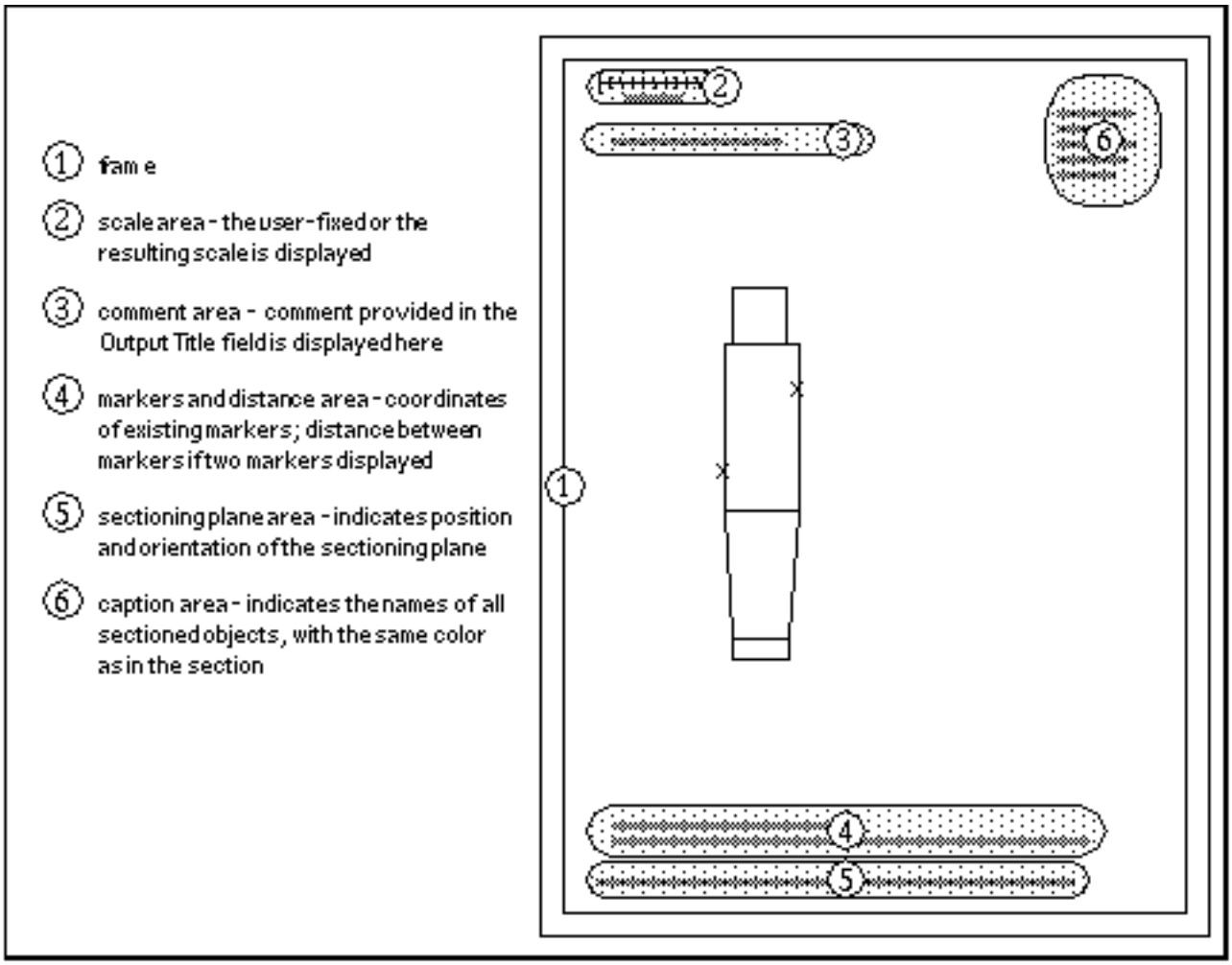
2. Type a file name in the Output Filename field. Full and relative UNIX names are legal. UNIX variable substitutions are not allowed. The tilde (~) is not a valid first character, but the following slash and dot combinations are legal: slash (/), dot-slash (./), and dot-dot-slash (../).
3. Press the Write HPGL button to write an HPGL file or press the Write Postscript button to write a PostScript file. An existing file of the same name is overwritten without a message.

You can print the contents of the Section window. The file that prints is a temporary file (`/usr/tmp/sectionoutput.tmp`) that is automatically deleted after printing. By default the `lpr` command prints the file, but you can specify a different command for PostScript files by defining the PVS_PRINT_COMMAND environment variable. You give the variable the value of an executable file such as a binary or script; no additional argument for the command is allowed. For example, the following line changes the default print command for PostScript files to `lp`:

```
setenv PVS_PRINT_COMMAND /usr/bin/lp
```

To print the contents of the Section window, press the Print HPGL or Print PostScript button.

In addition to the contents of the Section window (section, squaring, and dimensions), the following information appears in a printout:


1 frame
2 scale area - the user-fixed or the resulting scale is displayed
3 comment area - comment provided in the Output Title field is displayed here
4 markers and distance area - coordinates of existing markers; distance between markers if two markers displayed
5 sectioning plane area - indicates position and orientation of the sectioning plane
6 caption area - indicates the names of all sectioned objects, with the same color as in the section

Chapter 8

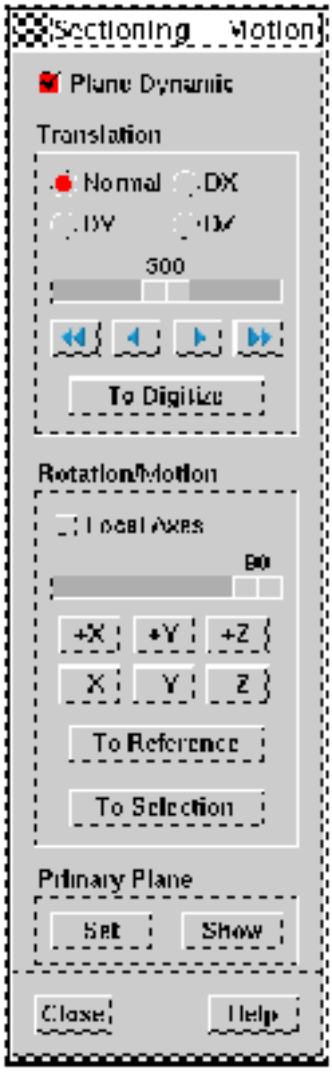
# Moving a Plane

This chapter describes additional operations that you can perform on the cutting and sectioning planes and includes information on the following:

- Getting Started
- Setting Plane Dynamics
- Translating the Plane
- Rotating the Plane
- Moving the Plane to a Reference
- Moving the Plane to the Selection List
- Setting and Retrieving the Primary Plane

# Getting Started

You can move a sectioning plane along a path or Gpath, and translate or rotate the plane. The graphical scene is automatically updated with every movement of the plane. To move the plane, select Motion from the Sectioning Control window. The Sectioning Motion window appears.



# Setting Plane Dynamics

You can set a sectioning plane to be dynamic. This allows you to move the sectioning plane along a path by itself or while it is attached to selected objects. This is useful in checking for interferences between moving and immobile components.

You can enable plane dynamics by selecting Plane Dynamic from the Sectioning menu or by pressing the Plane Dynamic button in the Sectioning Motion window.

Declaring the sectioning plane as dynamic does not interfere with the sectioning process. You can enable or disable the dynamic setting at any time without an adverse effect.

You begin to perform path traversal by a sectioning plane by selecting Item Dynamics from the Edit menu. Selecting Item Dynamics from the Edit menu allows you to move a sectioning plane in conjunction with selected items to which it is rigidly linked. You begin to perform rotation and translation by a sectioning plane by displaying the Sectioning - Motion dialog box. Some discussion of path traversal, relating specifically to sectioning planes, is included here. For detailed information about performing path traversal, refer to Chapter 3, “Using EPD.Visualizer Menus.”

When a sectioning plane is set to dynamic, two situations occur, depending on the contents of the selection list. These situations are the following:

- The selection list is empty
- Objects are selected

These two circumstances, as they relate to sectioning planes, are described in the following sections.

## Moving the Plane Without Objects Selected

When a sectioning plane is moved along a path, and the selection list is empty, the movement is the same as for any other moving object. As the plane moves along the path, the scene and section view reflect its motion. You can enable the movie recorder during the plane motion. At the end of each step, EPD.Visualizer refreshes the scene and the Section window according to the active parameters produced by the sectioning operations. The Section window, if visible, always shows the section as facing the plane. At any time, you can use the Sectioning - Motion dialog box to manually move the plane. For information about the sectioning operations and the Section window, refer to Chapter 7, “Controlling a Plane’s Display and Multiple Sectioning.”

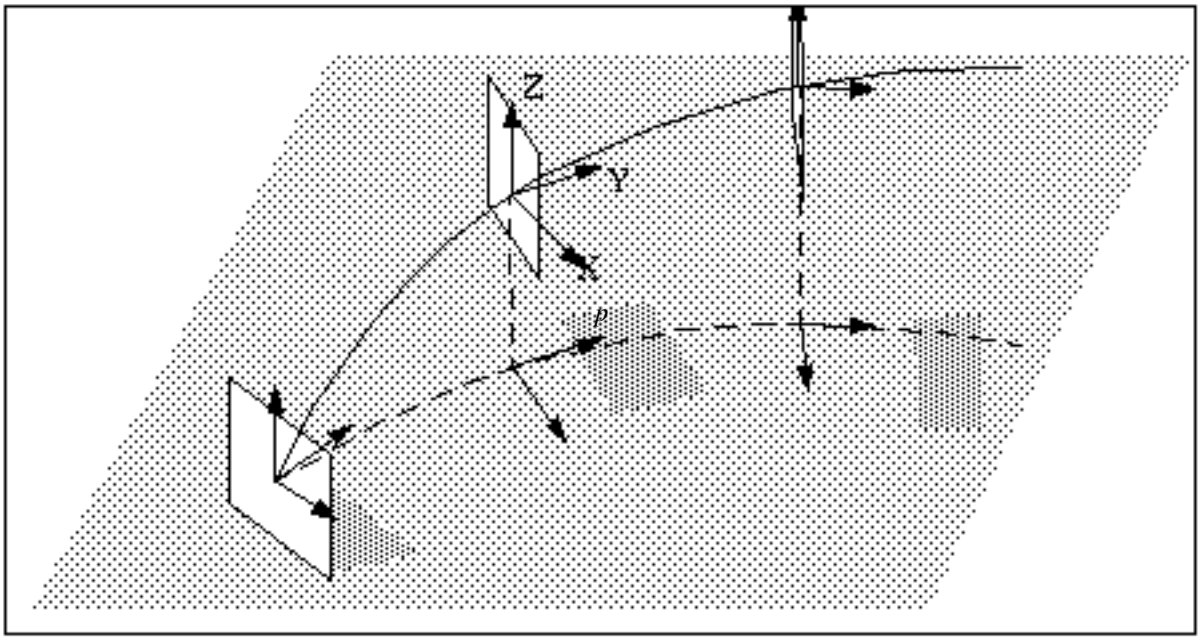
## Orienting

During the plane traversal, you can use both interactive and time delay methods. The orientation setting (Constant, Vertical, or To Path) allows you to manage the plane's behavior during the movement. The procedure for setting the interactive and time delay methods, and the orientation, are explained in detail in "Item Dynamics" on page 3-15. A description of the orientations, as they apply to a sectioning plane, follows:

- In a constant orientation, the plane's orientation remains the same during the motion. When performing item dynamics, you can display the plane's starting position and starting orientation at the beginning of the path. Also, you can adjust the starting position and starting orientation from the Sectioning Motion window.
- In a vertical orientation, the plane is considered attached to a referential moving along the path and able to rotate only around a vertical axis. The Z axis of the referential remains vertical during the motion. The Y axis of the referential is always tangent to the projection of the path in the XY plane. In the default position, which is set with the Start Position button in item dynamics, the plane is vertical and its normal vector is in a plane tangent to the path. Before, the traversal begins, you can change the starting position and starting orientation from the Sectioning Motion window.

  Figure 8-1 shows the vertical orientation of the plane as it traverses the path.

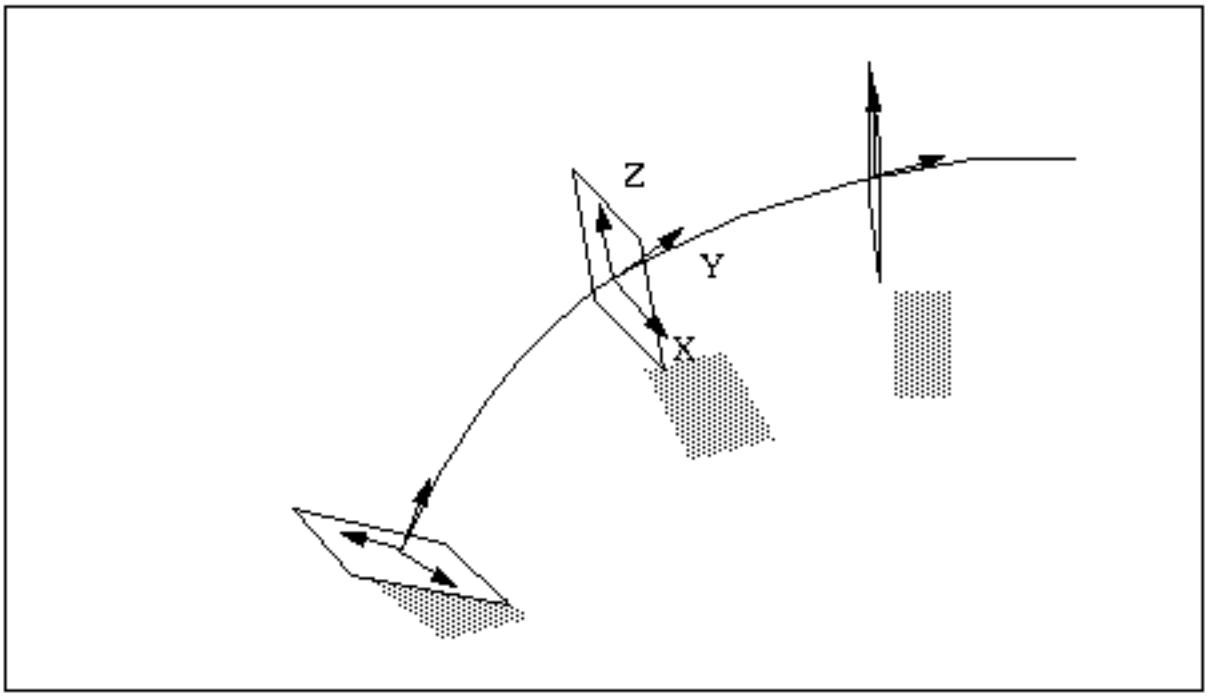
**Figure 8-1 Vertical Orientation of the Sectioning Plane**



- In orientation of the sectioning plane to the path, the plane is considered attached to a referential moving along the path and having its Y axis tangent to the path. You can change the starting position and orientation of the plane from the Sectioning Motion window.

  Figure 8-2 shows orientation of the plane to the path.

**Figure 8-2 Orientation of the Sectioning Plane to the Path**




## Translating and Rotating

If you want to translate or rotate the sectioning plane, without a selected object, you must use the Sectioning Motion window. If you attempt to translate or rotate the plane using item dynamics (specifically, by opening or using the Translate/Rotate Selection List dialog box) with an empty selection list, EPD.Visualizer displays the following error message:

```
The selection list is empty!
```

## Moving the Plane with Objects Selected

When the sectioning plane is dynamic and objects are selected, the plane attaches rigidly to the selection list and moving the list moves the plane. As with a plane that is not attached to objects, EPD.Visualizer uses the active sectioning parameters to refresh the scene and the Section window. You can use the Sectioning - Motion dialog box to change the plane position relative to the selected objects. This is particularly useful for adjusting the starting position and starting orientation. Positioning the plane precisely on the selection before moving them is possible with the feature discussed in “Moving the Plane to the Selection List” on page 8-10.

Selecting Item Dynamics from the Edit menu allows you to move a sectioning plane in conjunction with the selection list, to which it is rigidly linked, by specifying the plane’s movement in the Translate/Rotate Selection List dialog box. This is possible only if the selection list includes at least one item. As with path traversal, you can move the selected objects as you normally do, and the plane is rigidly linked to them. You can use the Sectioning Motion window to change the plane’s position independently of the selected objects.

# Translating the Plane

You can move the plane without changing its orientation. You can perform the movement step-by-step or by digitizing a vertex in the scene.

## Moving by Steps

When moving the plane by steps, you can perform the movement along the plane's normal vector or along one of the main axis (X, Y, or Z) of the scene. You can move the plane by steps in four ways. These are described in the following table:

| Option | Arrow Button Used | Description |
|---|---|---|
| Normal | Right and Left | Right arrow moves the plane to the right of the screen and the left arrow moves it to the left. If the plane is displayed almost horizontally on the screen, the right arrow moves it up and the left arrow moves it down. If the plane is almost parallel to the screen, the right arrow moves it towards you and the left arrow moves it away from you. |
| | Double | The length of the movement is the scale value. |
| | Single | The length of the movement is equal to one tenth of the scale value. |
| DY, DX, DZ | Right and Left | Right arrow moves the plane to the positive values. Left arrow moves the plane to the negative values. |

## Moving by Digitizing

You can move the plane to contain the digitized vertex. If you use the plane's size to display the plane, the plane's center is displayed on the digitized point.

# Rotating the Plane

You can rotate the plane around global or local axes. With the Local Axes toggle deactivated, you can rotate the plane around the global axes. With the Local Axes toggle activated, you can rotate the plane around its own axes. The local axes of the plane are displayed, attached to the plane's center. Regardless of whether or not the Local Axes toggle is activated, the local axes of the plane move with the plane.

Rotate the plane using the +X, +Y, +Z, -X, -Y, and -Z buttons. The value of the angle is indicated on the scale. You can modify the scale using the Scale Customization window. Click the right mouse button on the current scale value to display the Scale Customization window. The Scale Customization window is shown in "Specifying the Size" on page 7-3.

Following convention, the X and Y axes are in the plane while the Z axis is normal to it. The X axis is longer than the other two axes to aid in its identification. The X and Y axes are not always parallel to the plane's boundaries, because the plane's boundaries are always adjusted in order to place at least two parallel boundaries in one of the main planes of the scene: XY, YZ, or ZX.

# Moving the Plane to a Reference

You can move the plane to any visible reference. Press the To Reference button in the Sectioning - Motion dialog box's Rotation/Motion section and select a reference. The plane moves to the reference. The plane's local axes are always used in positioning it on a reference.

The PVS_SECTION_AXIS environment variable controls which reference axis is used as a normal in orienting the plane relative to a reference; the other axes of a selected reference are used to define the orientation of the local referential of the sectioning plane. The axis that precedes the axis defined as PVS_SECTION_AXIS defines the Y axis of the plane. The axis that follows the axis defined as PVS_SECTION_AXIS defines the X axis of the plane:

- When the X axis is defined as the PVS_SECTION_AXIS, a selected reference's Z axis defines the Y axis of the plane, and the reference's Y axis defines the X axis of the plane.
- When the Y axis is defined as the PVS_SECTION_AXIS, a selected reference's X axis defines the Y axis of the plane, and the reference's Z axis defines the X axis of the plane.
- When the Z axis is defined as the PVS_SECTION_AXIS, a selected reference's Y axis defines the Y axis of the plane, and the reference's X axis defines the X axis of the plane.

The default value of PVS_SECTION_AXIS is `X`. You can set PVS_SECTION_AXIS to any of the following values: `X`, `Y`, `Z`, `x`, `y`, or `z`.

# Moving the Plane to the Selection List

You can move the plane to the referential of the current selection list by pressing the To Selection button in the Sectioning - Motion dialog box's Rotation/Motion section. This feature is useful in preparing for the Plane Dynamic condition discussed in "Setting Plane Dynamics" on page 8-3.

The selection list has a local referential. The PVS_SECTION_AXIS environment variable controls which axis from the selection list's local referential defines the plane's normal. The other axes of the selection list define the orientation of the local referential of the sectioning plane. The default value of PVS_SECTION_AXIS is `X`. You can set PVS_SECTION_AXIS to any of the following values: `X`, `Y`, `Z`, `x`, `y`, or `z`.

# Setting and Retrieving the Primary Plane

You can use the Set option to set your primary plane. To retrieve the plane you have designated your primary plane, use the Show option.

Chapter 9

# Palettes

This chapter provides detailed information on the Tool palette and Walk-Through palettes (Move and Turn/Orbit palettes). The palettes are located vertically alongside the graphics window.

The topics discussed in this chapter are:

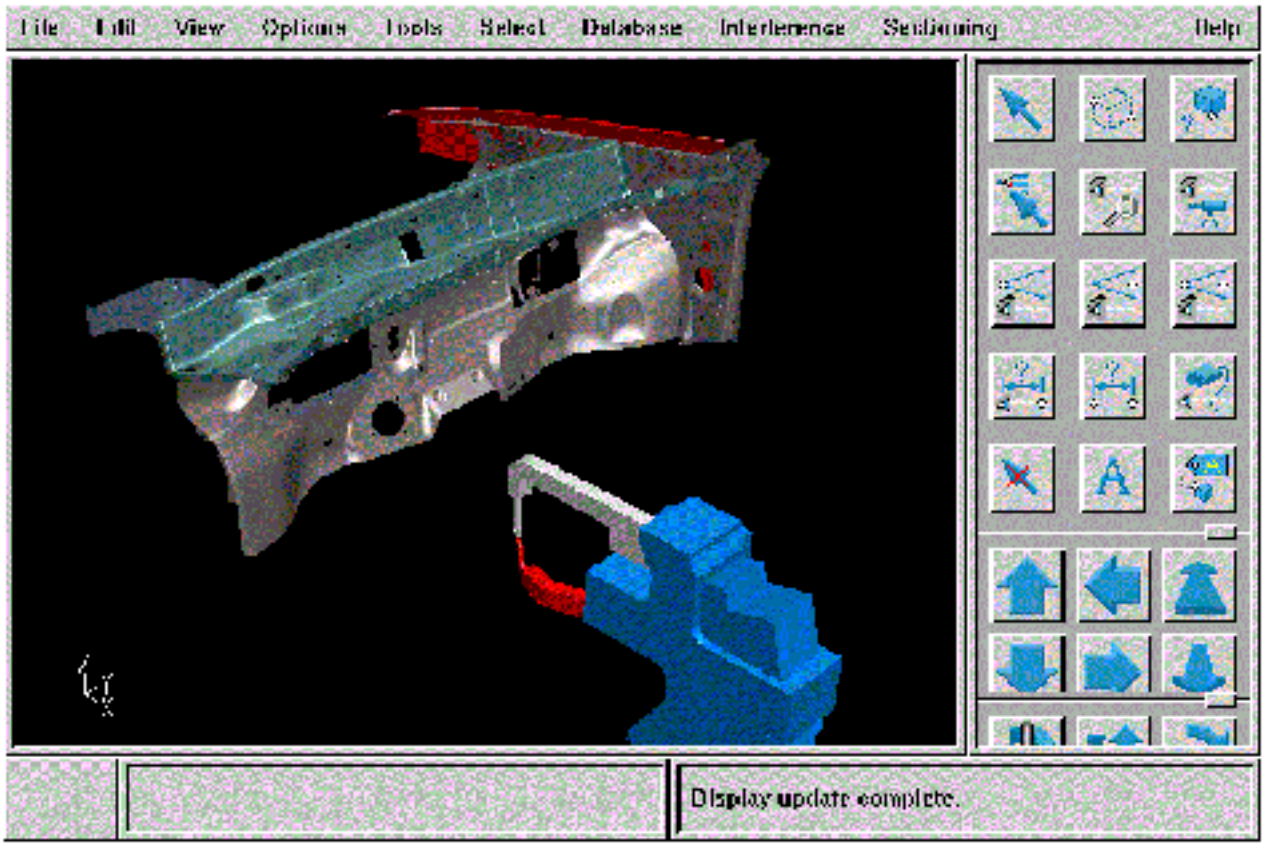
- Tool Palette
- Walk-Through Palettes
- Spaceball

You can access online help for the palettes by selecting Help > On Context. You can also point to the tools, buttons, or slider scales, and a brief description of each one displays in the prompt window. For basics on using the palettes, refer to Chapter 1, “Introduction to EPD.Visualizer.”

# Tool Palette

The Tool Palette contains tools that require you to perform an action with the mouse in the graphics window. Figure 9-1 shows the Tool Palette, which is the upper five rows of icons consisting of 15 selections:

**Figure 9-1 Tool Palette**




You can perform the following tasks with the tools. The tools are listed left-to-right and by row starting with the top row.

- Select items
- Select all items in a specified volume
- Retrieve information about selected items
- Tree select and scroll
- Permanent zoom by rectangular area
- Zoom a selected area
- Set a new camera position

- Set a new camera target
- Set a new camera position and camera target
- Measure the distance from the camera position to an item
- Measure the distance between two items
- Change the position or orientation of the camera using the mouse
- Clear current selection
- Create or edit text
- Manually or automatically tag objects

The following sections describe each tool in the Tool Palette from left to right.

## Select

Select single or multiple items by group or discipline.

To make a selection, use the following steps:

1. Select Select > Selection Mask. If you select Discipline or Group and click on an object, all objects within the discipline or group are selected.
2. Pick up the tool and move the pointer into the graphics window. You are prompted with

```
Select Items <By Discipline/By Group>:
Digitize items to add to the selection list...
```

To select multiple items, press the `Shift` key and continue to select items with the Select tool.

To select all items in the 3D scene, select Select > Build Selection List and add all Graphic Files to the selection list.

To deselect all items and clear the selection list, select Select > Deselect All, or press the left mouse button in empty space within the graphics window.

## Volume Select

Select all individual items in a specified volume.

Use the Volume Select tool to select individual items within a specified area rather than items by discipline or group.

To make a selection by volume, use the following steps:

1. Pick up the tool and move the pointer into the graphics window. You are prompted with:

```
Select By Volume:
Digitize first vertex point...
```

2. After you digitize an item, a digitize marker snaps to the nearest vertex and you are prompted with:

```
Select By Volume:
Digitize second vertex point...
```

3. After you digitize the second item, the selected items are highlighted and the following message displays:

```
Number of selected items = number
```

## Verify

Retrieve information about a selected item.

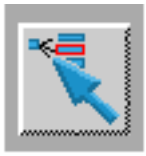
To verify items, use the following steps:

1. Pick up the tool and move the pointer into the graphics window. You are prompted with:

```
Verify Item:
Digitize item to verify...
```

2. After you digitize the item, the Verification dialog box appears.

## Tree Select and Scroll



Retrieve information about a selected item.

Select this command, then select a graphic component. In the EPD.Connect, the product structure tree scrolls to the items associated with the selected component. The selected component joins the selection list. To select multiple items, press the Shift key and continue to select items with the Tree Select and Scroll Tool.

## Permanent Zoom

Selecting this icon enables you to define a rectangular area to be a permanent close-up view of your 3D scene.

The Permanent Zoom moves the camera to the center of the selected rectangular area and attaches it to the existing geometry. Selectable geometry must exist under the center point of the rectangular area that you select.

For more information about this feature, refer to “Zoom” on page 9-6.

## Zoom

The Zoom tool enables you to define a rectangular area to be a temporary close-up view of your 3D scene.

A temporary close-up view is useful for briefly enlarging an area of your scene. The close-up view is similar to looking through binoculars.

To zoom a selected area, use the following steps:

1. Click on the tool and move the pointer into the graphics window. You are prompted with:

```
Zoom By Rectangular Area:
Press SELECT mouse button and drag...
```

2. As you start to drag the pointer, you are prompted with the following:

```
Drag pointer to define zoom area and release...
```

3. The selected area is magnified, and you are prompted with the following:

```
Click any mouse button to restore previous view...
```

## Set Camera Position

The Set Camera Position tool enables you to set a new position, or your eye point, for the camera.

To set a new Camera Position, use the following steps:

1. Click on the tool and move the pointer into the graphics window. You are prompted with the following:

```
Set Camera Position:
Digitize a vertex point...
```

2. After you digitize an item, the camera moves to that new point.

## Set Camera Target

The Set Camera Target tool enables you to set a point around which the orbit buttons operate.

To set a new Camera Target, use the following steps:

1. Click on the tool and move the pointer into the graphics window. You are prompted with the following:

```
Set Camera Target:
Digitize a vertex point...
```

2. After you digitize an item, the 3D scene adjusts to the new camera target.

## Set Camera Position and Target

The Set Camera Position and Target tool enables you to set a new Camera Position and Camera Target.

To set a new camera position and target, use the following steps:

1. Pick up the tool and move the pointer into the graphics window. You are prompted with the following:

```
Set Camera Position and Target:
Digitize Camera Position point...
```

2. After you digitize the Camera Position, you are prompted with the following:

```
Set Camera Position and Target:
Digitize Camera Target point...
```

## Measure Distance from Camera to Item

The Measure Distance from Camera to Item tool enables you to set the distance from the camera position to a selected item in the 3D scene.

To measure the distance between the camera and an item, use the following steps:

1. Click on the tool and move the pointer into the graphics window. You are prompted with the following:

```
Measure Distance From Camera To Item:
Digitize a vertex point...
```

2. After you digitize an item, a digitize marker snaps to the nearest vertex. The following message displays:

```
Distance to selected item = distance
```

   The distance displays in working units (such as feet, miles, or meters) defined in Options > Preferences.

## Measure Distance Between Items

The Measure Distance Between Items tool enables you to measure the distance between two items in the 3D scene.

The distance is measured by the straight line distance (calculated from vertex to vertex) and the delta xyz distances (expressed as the xyz component parts of the straight line distance).

To measure the distance between two items, use the following steps:

1. Click on the tool and move the pointer into the graphics window. You are prompted with the following:

```
Measure Distance Between Items:
Digitize first vertex point...
```

2. After you digitize the first item, a digitize marker snaps to the nearest vertex. You are prompted with the following:

```
Measure Distance Between Items:
Digitize second vertex point...
```

3. After you digitize the second item, the following message displays:

```
Distance between items = distance
dx = distance, dy = distance, dz = distance
```

The distance between the items is the straight line distance and is calculated in working units (such as feet, miles, or meters) defined in Options > Preferences.

The dx, dy, and dz distance is the delta xyz distance.

## Mouse Viewing

The Mouse Viewing tool enables you to change your view orientation using the mouse.

In addition to the Walk-Through buttons, you can use the mouse to change your view orientation based on the specified target (camera or 3D scene). Each time you move the mouse a specific number of pixels, the display repaints. The target and pixel number are specified in Preferences in the Options menu.

During all orbit or move operations, the status window indicates the distance or the number of degrees the mouse has moved. Once you release the mouse button, this number is reset to zero.

To set Options > Preferences, use the following steps:

1. Specify the target with one of the following options:
   - Camera — Moves the camera position in the direction of the mouse pointer.
   - Scene — Moves the 3D scene in the direction of the mouse pointer. The camera remains stationary.
2. Specify the number of pixels in the Threshold slider scale.

To orbit right, left, over, and under, use the following steps:

This function is equivalent to the Orbit Left, Orbit Right, Orbit Over, and Orbit Under buttons on the Turn/Orbit Palette. It orbits the position of the camera or the

3D scene to the left, right, over, or under the vertical axis of the 3D scene. To use the mouse to orbit the camera or the 3D scene, follow these steps:

1. Click the tool and move the pointer into the graphics window.
2. Hold down the left mouse button and move the mouse left, right, up, or down to orbit.

   What you see as you orbit is relative to the camera target, or relative to the 3D scene. To change the camera target, click on the Set Camera Target tool and select a new target in the graphics window.

   You cannot orbit upside down. The number of degrees you orbit is determined by the setting on the Turn/Orbit Angle slider scale.

To move up, down, left, and right, use the following steps:

This function is equivalent to the Move Up, Move Down, Move Left, and Move Right buttons on the Move Palette. It moves up, down, left, or right relative to the coordinate system of the graphics window. To use the mouse to move the camera or the 3D scene, follow these steps:

1. Click on the tool and move the pointer into the graphics window.
2. Hold down the middle mouse button and move the mouse left, right, up, or down.

   The number of units you move is determined automatically, based on the size of the camera opening. The size of the camera opening is determined by the perspective angle.

To move forward and back, use the following steps:

This function is equivalent to the Move Forward and Move Back buttons on the Move Palette. It moves forward or back relative to the coordinate system of the graphics window. To use the mouse to move the camera or the 3D scene forward or back, follow these steps:

1. Click on the tool and move the pointer into the graphics window.
2. Hold down the `Shift` key plus the middle mouse button and move the mouse up to move forward and down to move backward.

   If you move beyond the camera target, a new camera target is automatically set. The number of units you move is determined by the units set on the Move Distance slider scale.

To turn right and left, use the following steps:

This function is equivalent to the Turn Right and Turn Left buttons on the Turn/Orbit Palette. It turns the camera or 3D scene to the right or left. To use the mouse to turn the camera or the 3D scene right or left, follow these steps:

1. Click on the tool and move the pointer into the graphics window.
2. Hold down the right mouse button and move the mouse right or left.

   The number of degrees you turn is determined by the setting on the Turn/Orbit Angle slider scale.

To look up and down, use the following steps:

This function is equivalent to the Look Up and Look Down buttons on the Turn/Orbit Palette. It enables you to look up or look down the vertical axis of the 3D scene. To use the mouse to look up or down, follow these steps:

1. Pick up the tool and move the pointer into the graphics window.
2. Hold down the right mouse button and move the mouse up or down.

   The number of degrees you look up or down is determined by the setting on the Turn/Orbit Angle slider scale.

## Deselect

Selecting this icon clears the current selection list. Clicking on this icon does not inactivate the currently active tool.

## Text

Create or edit text at a specific point in the graphics window.

To enter new text, use the following steps:

1. You can either use the default text properties or set new text properties by selecting Edit > Text Properties.
2. Click on the tool and move the pointer into the graphics window. You are prompted with:

```
Insert Text:
Digitize text location...
```

3. Digitize a location and the Text Editor with the xyz coordinates of the digitize displays.

Please note: Before you enter text, you can select a different location without closing the Text Editor.

4. Click in the text box to enter text.
5. Click OK and the text appears.

To edit existing text, use the following steps:

1. Select the text you want to edit using the Text tool.
2. Edit the text in the text box.
3. Click OK and the new text displays.

To change text properties of existing text:

1. Select the text with the Select tool.
2. Select Edit > Text Properties.
3. Set the new text properties and click Apply. The text displays with the new properties.

# Object Tagging

The object tagging tool enables you to manually or automatically tag objects.

## Manually Tag Objects

To manually tag objects, use the following steps:

1. Set Database > Tagging to Manual.
2. You can use either the default text properties or set new text properties by selecting Edit > Text Properties.
3. Click on the tool and move the pointer into the graphics window. You are prompted with:

```
Tag Object Manually:
Digitize object to tag...
```

4. Digitize the vertex of an object and the Text Editor displays with the xyz coordinates of the object.

Please note: Before you enter text, you can select a different location without closing the Text Editor.

5. Click in the text box to enter text.
6. Click OK and the text displays.

## Automatically Tag Objects

To automatically tag objects with a tag stored in the database, use the following steps:

1. Select Database > Open Database to open a database file.
2. Set Database > Tagging to Automatic.

3. You can use either the default text properties or set new text properties by selecting Edit > Text Properties.
4. Click on the Object Tagging tool and move the pointer into the graphics window. You are prompted with the following:

```
Tag Object Automatically:
Digitize object to tag...
```

5. After you digitize the vertex of an object, the predefined tag from the database appears.

# Walk-Through Palettes

Use the walk-through palettes to change the orientation and position of the camera. You can perform the following operations using the walk-through buttons and slider scales:

- Move in a linear direction by a specific number of units
- Move in an angular direction by a specific number of degrees
- Level the camera position or camera target to a horizontal plane

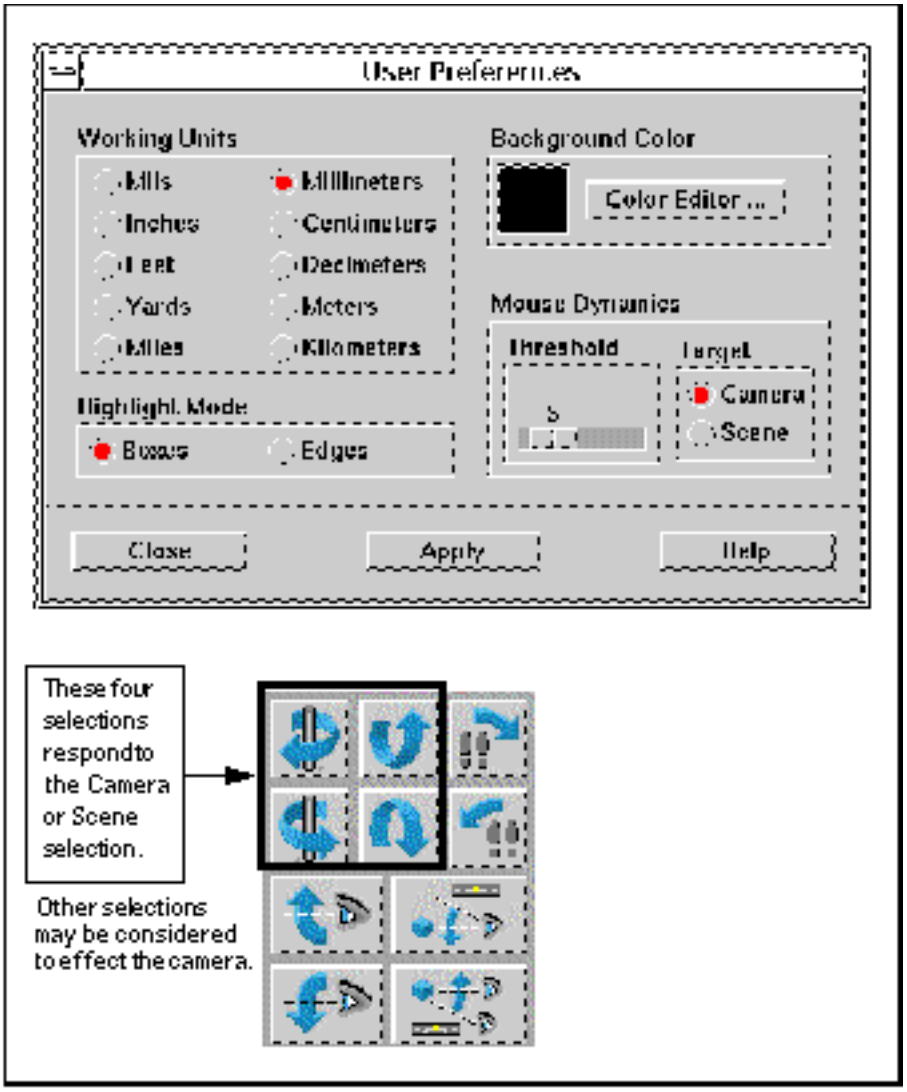
You can control the rendering of the 3D scene while you use the walk-through buttons. The following describes these methods:

- To move the camera rapidly through the 3D scene without waiting for a complete rendering of all graphics, click any of the walk-through buttons before the 3D scene completely displays.
- To suppress the rendering of the 3D scene while using any of the walk-through buttons, hold down the `Shift` key while clicking the mouse button.
- To change the Move Distance or Turn/Orbit Angle during rendering of the 3D scene, click the slider scales.

The selection in the Target area of the User Preferences dialog box determines the effect of four walk-though palettes tools identified in the following figure. The four tools move either the camera or the scene, depending on your selection in the Target area of the User Preferences dialog box. You display the User Preferences dialog box by selecting Options > Preferences. The selections in the Target area of the User Preferences dialog box are the following:

- Camera — Pressing the Camera button specifies that the four walk-though palettes selections with definable focus move the camera. For example, in this mode, selecting the left arrow moves the scene to the right as an effect of the camera moving to the left. This is the default mode.
- Scene — Pressing the Scene button specifies that the four walk-though palettes selections with definable focus move the scene. For example, in this

mode, selecting the left arrow moves the scene to the left. This mode can be specified on the command line by the option `-scene`.




The following sections describe the walk-through palettes (the Move Palette and the Turn/Orbit Palette).

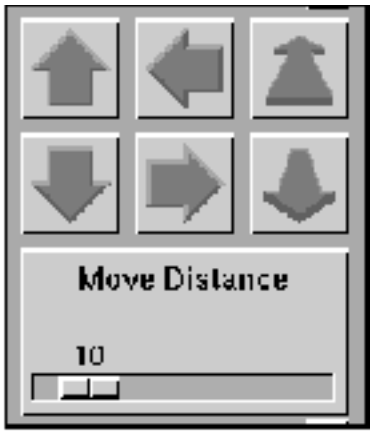
## Move Palette

The Move Palette contains buttons that enable you to move in a linear direction. For each of the tools in the Move Palette, the number of units you move is determined by the units set on the Move Distance slider scale. Refer to the “Move Distance Slider Scale” on page 9-18 for more information.

You can specify the working units by selecting Options > Preferences.

Figure 9-2 shows the Move Palette, including the move buttons and the Move Distance slider scale:

**Figure 9-2 Move Palette**




## Move Up

The Move Up tool enables you to move the 3D scene up relative to the coordinate system of the graphics window.

## Move Down

The Move Down tool enables you to move the 3D scene down relative to the coordinate system of the graphics window.

## Move Left

The Move Left tool enables you to move the 3D scene left relative to the coordinate system of the graphics window.

## Move Right

The Move Right tool enables you to move the 3D scene right relative to the coordinate system of the graphics window.

## Move Forward

The Move Forward tool enables you to move the 3D scene forward relative to the coordinate system of the graphics window.

If you move beyond the camera target, a new camera target is automatically set.

## Move Back

The Move Back tool enables you to move the 3D scene back relative to the coordinate system of the graphics window.

Each time you click the button, you move away from the 3D scene and more of the scene comes into view.

## Move Distance Slider Scale

Set the number of units the camera position moves in a linear direction.

To customize the slider scale, click your right mouse button on the slider scale value. For more information on customizing the slider scale, refer to “Slider Scales” on page 1-13 in Chapter 1, “Introduction to EPD.Visualizer.”

To set the working units select Options > Preferences. For more information on setting the working units, refer to “Options Menu” on page 3-73.

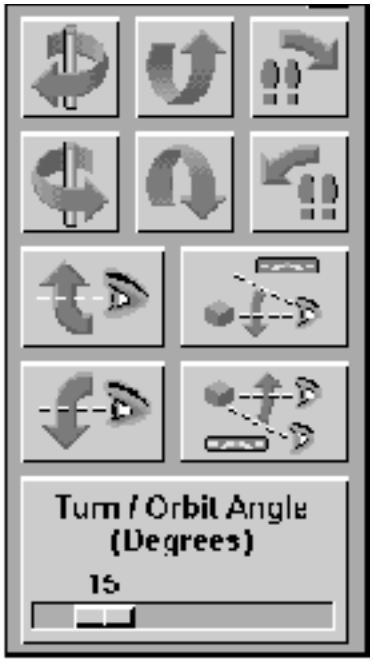
## Turn/Orbit Palette

The Turn/Orbit Palette contains buttons that enable you to move in an angular direction. For each of the tools in the Turn/Orbit Palette, what you see as you orbit is relative to the camera target. To change the camera target, click on the Set Camera Target tool and select a new target point in the graphics window.

The number of degrees you orbit, turn, or look up and down, is determined by the setting on the Turn/Orbit Angle slider scale. Refer to “Turn/Orbit Angle Slider Scale” on page 9-22 for more information. You cannot orbit upside-down.

Figure 9-3 shows the Turn/Orbit Palette, including the orbit, look, turn, and level buttons and the Turn/Orbit Angle slider scale:

**Figure 9-3 Turn/Orbit Palette**



## Orbit Left

The Orbit Left tool enables you to orbit the camera position to the left around the vertical axis of the 3D scene.

## Orbit Right

The Orbit Right tool enables you to orbit the camera position to the right around the vertical axis of the 3D scene.

## Orbit Over

The Orbit Over tool enables you to orbit the camera position over the vertical axis of the 3D scene.

## Orbit Under

The Orbit Under tool enables you to orbit the camera position under the vertical axis of the 3D scene.

## Turn Right

The Turn Right tool enables you to turn the camera target to the right.

## Turn Left

The Turn Left tool enables you to turn the camera target to the left.

## Look Up

The Look Up tool enables you to look up the vertical axis of the 3D scene.

## Look Down

The Look Up tool enables you to look up the vertical axis of the 3D scene.

## Level Off to Camera Position

The Look Up tool enables you to look up the vertical axis of the 3D scene.

## Level Off to Camera Target

The Level Off to Camera Target tool enables you to level off to the camera target. As you level off, either up or down, you maintain the camera target.

## Turn/Orbit Angle Slider Scale

Set the number of degrees the camera position moves in an angular direction.

To customize the slider scale:

Click your right mouse button on the slider scale value. For more information on customizing the slider scale, click the HELP button or refer to “Slider Scales” on page 1-13.

# Spaceball

The Spaceball interface is supported for EPD.Visualizer, but information about it is included here for users who want to evaluate and test the product for use with EPD.Visualizer.

## Setting Up Spaceball

You must complete the following steps before using Spaceball with EPD.Visualizer.

1. Attach Spaceball to an RS232 port on the back of the workstation.
2. Connect the A/C adapter to the Spaceball cable and plug the power supply into an A/C outlet. The Spaceball should beep twice.
3. Create a symbolic link called `/dev/spaceball` by logging in as root and entering the following:

   ```
   ln -s device_file /dev/spaceball
   ```

   Where device_file is the name of the device file for the serial port where Spaceball is connected (such as `/dev/tty00`).

   For detailed instructions for each workstation manufacturer, please refer to your Spaceball Manual.
4. Start the Spaceball daemon located in the `/usr/apl/CVpvs/bin` directory by entering the following:

   ```
   cd ${VIS_HOME}/bin
   ./sballd_XCM&
   ```

   For IRIX 6.5, AIX 4.2.1, and OSF 3.2c, use the following:

   ```
   cd ${VIS_HOME}/bin
   ./sballd_XCM -GL&
   ```

   The Spaceball beeps twice if all is well.

## Spaceball Button Summary

- Button 1 — Turns rotations on. For Object Edit mode, toggles translations on or off.
- Button 2 — Turns translations on. For Object Edit mode, toggles rotations on or off.
- Button 4 — Turns Camera mode on. If you are already in Camera mode, it toggles between Camera Orbit and Camera Walk-Through modes.
- Button 5 — Decreases sensitivity of Spaceball.

- Button 6 — Increases sensitivity of Spaceball.
- Button 7 — Turns Object Edit mode on.
- Button 8 — Rezeros Spaceball.
- Pick button (located on the back side of the ball) — Restores the primary view when you are in Camera mode.

## Using the Spaceball with EPD.Visualizer

To enable Spaceball capabilities, execute EPD.Visualizer with the `-spaceball` option. The EPD.Visualizer Spaceball interface supports Camera and Object Edit modes.

### Camera Mode

Camera mode enables you to toggle between Camera Orbit mode and Camera Walk-Through mode. These Camera modes are toggled by pressing Spaceball button 4. The current mode is displayed in the right message area of the EPD.Visualizer interface.

In Camera Orbit mode you can twist the Spaceball on its axis to orbit the camera about the vertical axis of the 3D scene. If you tilt the Spaceball forward or backward, the camera orbits up and down respectively.

In Camera Walk-Through mode, twisting the Spaceball left or right turns the camera to the left or right respectively. Pressing the Spaceball pick button restores the primary view when you are in Camera mode.

### Object Edit Mode

To enter Object Edit mode, press Spaceball button 7. Objects must be on a selection list to edit them. In Object Edit mode you can toggle translations and rotations on or off by pressing Spaceball buttons 1 and 2.

Pulling up or down on the Spaceball causes selected objects to move up or down the vertical axis. You can move objects along an xy axis by pushing or pulling the Spaceball in a direction defined by the axis display.

To zoom in or out, push or pull the Spaceball towards or away from the screen. Rezero the Spaceball by pressing Spaceball button 8.

Spaceball buttons 5 and 6 decrease or increase the sensitivity of the Spaceball, respectively.

Appendix A

# Database Interface

Visualizer has an SQL interface that enables you to connect to the Oracle Database Management System (DBMS) in addition to opening an ASCII database file. This appendix provides information on how to create an ASCII database and construct object tables interfacing the DBMS. The following topics are discussed:

- Creating an ASCII Database
- DBMS SQL Interface
- DBMS Demo

After you create a database, you can use it for the following purposes:

- Obtain detailed engineering data about an object using the Verify tool.
- Tag an object with the tag stored in the database file using the Object Tagging tool after selecting Database > Tagging > Automatic.
- Send SQL statements to select or edit data in DBMS tables. Objects selected from the DBMS can be highlighted, zoomed or added to the selection list.

# Creating an ASCII Database

A database used with Visualizer can have an unlimited number of records, with each object record consisting of one line.

To create an ASCII database, the following flat files are required:

- Database File — This file is generated using either your database program or another method to collect the detailed data. Each database record must include the following information:
  - Object ID (`OBJECT_ID`)
  - Graphic file name (`MODEL`)
  - Object type (`OBJECT_TYPE`)

  The database file can contain records from various disciplines, such as piping, steel, or equipment.
- Database Definition File — This file defines all object types and object record formats in the database file. After you create the database file, you must define the record structure of objects in the definition file. The definition file describes object records from various disciplines.

  When you open a database file using the Database menu items, Visualizer uses the information in the database definition file to open it. When an object is selected for verification or tagging, the `OBJECT_ID` and `MODEL` (graphic file) names are sent to the database program. The database program uniquely identifies the database record with `OBJECT_ID` and `MODEL` (graphic file) in the database file and displays the object record or object tag field(s).

# Example of a Database Definition File

The following example shows a sample database definition file, and the structure of the file.

```
#DEFINE

!-----------------------------------------------------------------
-!
! Any character after an exclamation point (!) is regarded as a   !
! comment and is ignored by the database program. Blank lines     !
! are also ignored. Field names are not case-sensitive.           !
!-----------------------------------------------------------------
-!

   #DEFINE_DB_FILE
FILE_NAME = /hp/moe/b/proj/cvpvs/data/mckee.db
#END

!-----------------------------------------------------------------
-----!
! The "#DEFINE_REQUIRED_FIELDS" section must appear before the
!
! "#DEFINE_OBJECT" section. In this section, OBJECT_ID, MODEL, and
!
! OBJECT_TYPE fields are defined by DEFINE_FIELD statements.
!
!                                                                 !
! Only one (1) "#DEFINE_REQUIRED_FIELDS" section is required per
!
! definition file.                                                !
!                                                                 !
! OBJECT_ID, MODEL, and OBJECT_TYPE data must appear in every
record  !
! in the database file. Define other fields in the "#DEFINE_OBJECT"
!
! section.                                                        !
!                                                                 !
! A DEFINE_FIELD statement has the following syntax:
!
!                                                                 !
! DEFINE_FIELD (field_name, starting_position, ending_position)
!
!                                                                 !
! A comma (,) is a separator.
!
!                                                                 !
! OBJECT_ID, MODEL, and OBJECT_TYPE field names are fixed for all
!
! object types and should not be changed. However, you can choose
!
! starting and ending column positions for these fields to
customize  !
```

```
! your object record. Refer to the #DEFINE_OBJECT section if you
!
! want to designate any of these fields for automatic tagging.
!
!                                                                    !
!--------------------------------------------------------------------
-----!

   #DEFINE_REQUIRED_FIELDS
DEFINE_FIELD (OBJECT_ID,  1, 16)
DEFINE_FIELD (MODEL,      17, 44)
DEFINE_FIELD (OBJECT_TYPE,73, 88)
#END

!--------------------------------------------------------------------
-----!
! One #DEFINE_OBJECT section is required for each object type.
!
!!
! "NAME = <object name>" must appear before any "DEFINE_FIELD" in
      !
! the! "#DEFINE_OBJECT" section.
      !
!!! Additional fields can be defined by DEFINE_FIELD statements.
      !
! !
! A DEFINE_FIELD statement has the following syntax:
! !
! DEFINE_FIELD (field_name, starting_position, ending_position)
!
! !
! A comma (,) is a separator.
!
! The field to be displayed by automatic tagging is defined by a
      !
! DEFINE_AUTO_TAG statement (optional). If required fields
(OBJECT_ID,!
! MODEL, and OBJECT_TYPE) must be defined for tagging, define them
!
! in this section by DEFINE_AUTO_TAG statement
!
!                                                                   !
! A DEFINE_AUTO_TAG statement has the following syntax:
      !
!                                                                   !
! DEFINE_AUTO_TAG = <field name>                                    !
!                                                                   !
! The "field_name" must be defined by a DEFINE_FIELD statement
BEFORE       !
! it is defined for automatic tagging.                             !
!                                                                   !
!--------------------------------------------------------------------
-----!
```

```
#DEFINE_OBJECT
      OBJECT_TYPE = PLPIPE
      DEFINE_FIELD (SUBMODEL,  45,  72)
      DEFINE_FIELD (ESID,      89,  94)
      DEFINE_FIELD (NPS1,      95, 102)
      DEFINE_FIELD (NPS2,     103, 110)
      DEFINE_FIELD (NPS3,     111, 118)
      DEFINE_FIELD (NPS4,     119, 126)
      DEFINE_FIELD (SPEC_MCC, 127, 138)
      DEFINE_FIELD (DESC,     139, 162)
      DEFINE_FIELD (MATERIAL, 163, 178)
      DEFINE_FIELD (STOCK_ID, 179, 190)
      DEFINE_FIELD (line_no,  191, 222)

      DEFINE_AUTO_TAG = LINE_NO

#END

#DEFINE_OBJECT
      OBJECT_TYPE = PLFITTING
      DEFINE_FIELD (SUBMODEL,  45,  72)
      DEFINE_FIELD (ESID,      89,  94)
      DEFINE_FIELD (NPS1,      95, 102)
      DEFINE_FIELD (NPS2,     103, 110)
      DEFINE_FIELD (NPS3,     111, 118)
      DEFINE_FIELD (NPS4,     119, 126)
      DEFINE_FIELD (SPEC_MCC, 127, 138)
      DEFINE_FIELD (DESC,     139, 162)
      DEFINE_FIELD (MATERIAL, 163, 178)
      DEFINE_FIELD (Stock_id, 179, 190)
      DEFINE_FIELD (LINE_NO,  191, 222)

      DEFINE_AUTO_TAG = OBJECT_TYPE
      DEFINE_AUTO_TAG = LINE_NO

#END

#DEFINE_OBJECT
      OBJECT_TYPE = SLLOBJECT
      DEFINE_FIELD (SUBMODEL,  45,  72)
      DEFINE_FIELD (ESID,      89,  94)
      DEFINE_FIELD (ITEMNAME,  95, 106)
      DEFINE_FIELD (LENGTH,   107, 114)
      DEFINE_FIELD (SHAPE,    115, 138)
      DEFINE_FIELD (STOCK_ID, 139, 150)
      DEFINE_FIELD (SDESC,    151, 162)
      DEFINE_FIELD (LDESC,    163, 226)

      DEFINE_AUTO_TAG = SDESC

#END

#DEFINE_OBJECT
      OBJECT_TYPE = SLPOBJECT
```

```
      DEFINE_FIELD (SUBMODEL,  45,  72)
      DEFINE_FIELD (ESID,      89,  94)
      DEFINE_FIELD (ITEMNAME,  95, 106)
      DEFINE_FIELD (LENGTH,   107, 114)
      DEFINE_FIELD (SHAPE,    115, 138)
      DEFINE_FIELD (STOCK_ID, 139, 150)
      DEFINE_FIELD (SDESC,    151, 162)
      DEFINE_FIELD (LDESC,    163, 226)
      DEFINE_AUTO_TAG = STOCK_ID
#END

#DEFINE_OBJECT
      OBJECT_TYPE = EQUIPMENT
      DEFINE_FIELD (SUBMODEL, 45,  72)
      DEFINE_FIELD (EQTYPE,   89, 104)
      DEFINE_FIELD (EQDESC,  105, 128)
#END

#DEFINE_OBJECT
      OBJECT_TYPE = PLX99
      DEFINE_FIELD (SUBMODEL,  45,  72)
      DEFINE_FIELD (ESID,      89,  94)
      DEFINE_FIELD (NPS1,      95, 102)
      DEFINE_FIELD (NPS2,     103, 110)
      DEFINE_FIELD (NPS3,     111, 118)
      DEFINE_FIELD (NPS4,     119, 126)
      DEFINE_FIELD (SPEC_MCC, 127, 138)
      DEFINE_FIELD (DESC,     139, 162)
      DEFINE_FIELD (MATERIAL, 163, 178)
      DEFINE_FIELD (STOCK_ID, 179, 190)
      DEFINE_FIELD (LINE_NO,  191, 222)

      DEFINE_AUTO_TAG = OBJECT_ID
#END

#END_DEFINE
```

# Structure of Database Definition File

A definition file starts with #DEFINE and ends with #END_DEFINE statements, including three sections in this order:

- #DEFINE_DB_FILE — The complete path name to the database file is defined. This section is defined only once for each definition file.
- #DEFINE_REQUIRED_FIELDS — The fields required to be in each database record are defined (object ID, model, and object type). This section is defined only once for each definition file.
- #DEFINE_OBJECT — The remaining fields for each database record type (object type) are defined. One #DEFINE_OBJECT section is required for each object type.

Each section ends with an #END statement. Text after an exclamation point (!) is regarded as comments and is ignored. These three sections are described in more detail as follows.

## #DEFINE_DB_FILE Section

In the #DEFINE_DB_FILE section, you specify the database file name with the complete path name. The following is an example of this section:

```
#DEFINE_DB_FILE
FILE_NAME = /moe/b/proj/cvpvs/data/mckee.db
#END
```

## #DEFINE_REQUIRED_FIELDS Section

There is only one #DEFINE_REQUIRED_FIELDS section for each definition file. In the #DEFINE_REQUIRED_FIELDS section, you define the OBJECT_ID, MODEL, and OBJECT_TYPE fields using DEFINE_FIELD statements. (The OBJECT_ID for DIMENSION III is described in the following section.) These field names are fixed for all object types and should not be changed. However, you can choose where these fields start and end in the object records.

OBJECT_ID, MODEL, and OBJECT_TYPE are defined by DEFINE_FIELD statements. A DEFINE_FIELD statement has the following syntax, with the comma used as a separator:

```
DEFINE_FIELD (field_name, starting_position, ending_position)
```

Only the OBJECT_ID, MODEL, and OBJECT_TYPE fields are defined in the #DEFINE_REQUIRED_FIELDS section:

- The OBJECT_ID refers to the object ID associated with each object in the Visualizer graphic file.
- The MODEL is the name of the Visualizer graphic file in which the object resides (without an extension).
- OBJECT_ID and MODEL fields uniquely identify each object in the database.

Define the remaining fields in the object database in the DEFINE_OBJECT section.

The OBJECT_TYPE determines which of the #DEFINE_OBJECT sections is used for describing data from each record. This is the name that is used in the first line of each #DEFINE_OBJECT section. For example, in "Example of a Database Definition File" on page A-3 in the #DEFINE_OBJECT section, the name is PLPIPE.

## OBJECT_ID FOR DIMENSION III Items (Objects)

Every item in a DIMENSION III model is uniquely described in the DIMENSION III database by an XIDPTR (extended identification pointer). An XIDPTR is a combination of two integer numbers; the first number is called the Model ID pointer (MID) and the second number is called the Element ID pointer (EID).

The following is an example of an XIDPTR:

```
-32765 -32711
```

The Model ID pointer (MID) is -32765, and the Element ID pointer (EID) is -32711.

The Model ID pointer describes a particular submodel within the parent model, so all the elements within a particular submodel have the same MID. If an element has an MID equal to -32768, then that element belongs to the parent model. Submodels begin with the MID -32767 and are incriminated by one for each additional submodel.

The Element ID pointer identifies a particular item within a model or submodel. Every item in a particular model or submodel has a unique EID. Submodels might (and probably will) contain elements with the same EID as elements within other submodels. However, with the combination of the unique Model ID and the Element ID (unique within a model), we can uniquely describe each element with the XIDPTR.

The XIDPTR of an element is verified within a DIMENSION III model with the DAL function AECXIDSCL, as follows:

```
OUT = AECXIDSCL IS
```

Where:

- IS = an item select of the element
- OUT = a list of two numbers; the first number is the Model ID pointer and the second number is the Element ID pointer

Please note: In DIMENSION III, equipment consists of a set of items. Each item has its own XIDPTR. Therefore, you must create a record in the database for each item (XIDPTR) that you want to verify in Visualizer.

In a Visualizer database file, OBJECT_ID is created from XIDPTR. OBJECT_ID is a long integer (4 bytes) that consists of two short integers (each 2 bytes) of the XIDPTR. This conversion can be described by FORTRAN and C languages as follows:

FORTRAN:

```
INTEGER*2 XIDPTR(2)
INTEGER*4 OBJECT_ID

EQUIVALENCE (OBJECT_ID, XIDPTR)
```

C:

```
union {
     short int   xidptr[2];
     long int   object_id;
}objectRecord;
```

The Visualizer database uses a combination of the OBJECT_ID and model viewfile names to uniquely describe an item in Visualizer.

## #DEFINE_OBJECT Section

In the #DEFINE_OBJECT section, the remaining fields for each object are defined. One #DEFINE_OBJECT section is required for each object type.

The first line in the #DEFINE_OBJECT section must be an OBJECT_TYPE=<object type name> statement. This statement determines which of the #DEFINE_OBJECT sections will be used for describing data from each record. The object type name must be the same name as specified for the OBJECT_TYPE field in the object record.

Database fields other than OBJECT_ID, MODEL, and OBJECT_TYPE are defined by DEFINE_FIELD in the #DEFINE_OBJECT section.

Automatic tags are defined by the DEFINE_AUTO_TAG statement. You must define the field_name with a DEFINE_FIELD statement before it is defined for automatic tagging.

Below is an example of a #DEFINE_OBJECT section:

```
#DEFINE_OBJECT
      OBJECT_TYPE = PLPIPE
      DEFINE_FIELD (SUBMODEL, 45, 72)
      DEFINE_FIELD (ESID,     89, 94)
      DEFINE_FIELD (NPS1,     95, 102)
      DEFINE_FIELD (NPS2,     103, 110)
      DEFINE_FIELD (NPS3,     111, 118)
      DEFINE_FIELD (NPS4,     119, 126)
      DEFINE_FIELD (SPEC_MCC, 127, 138)
      DEFINE_FIELD (DESC,     139, 162)
      DEFINE_FIELD (MATERIAL, 163, 178)
      DEFINE_FIELD (STOCK_ID, 179, 190)
      DEFINE_FIELD (LINE_NO,  191, 222)
      DEFINE_AUTO_TAG = LINE_NO
#END
```

NO TAG is a sample database file (not to scale) described by the database definition file. This is an ASCII file that is created by your database program or another method.

# DBMS SQL Interface

The DBMS SQL interface lets you connect to a Database Management System to obtain essential engineering data.

When an object is digitized, the Verify tool queries the DBMS for database information. The Verify tool displays object data that is retrieved from the connecting DBMS in the Object Verification dialog.

The Visualizer DBMS interface requires that you construct several tables. Once these tables are constructed, you can retrieve an SQL query statement. This statement is used to retrieve specific verification data for the digitized object.

## Verification Tables

There are four tables required by the interface:

- `PVS_OBJECT` (See Table A-1 on page A-12)
- `PVS_QUERY` (See Table A-2 on page A-12)
- `PVS_QUERY_MODE` (See Table A-3 on page A-13)
- `PVS_SQL_STMT` (See Table A-4 on page A-13)

The first table is called the `PVS_OBJECT` table. This table contains the following fields:

- `OBJECT_ID` — Object identification that corresponds to the `object_ id` attribute in Visualizer
- `FILE_NAME` — Graphics file name where object resides
- `OBJECT_TYPE` — A specific type of an object (for example, valve, pipe, steel). This field can be `NULL`. A null object type indicates that it is not needed when searching for a query statement.

Below is an example of this table:

**Table A-1 Sample PVS_OBJECT table**

| OBJECT_ID | FILE_NAME | OBJECT_TYPE |
|---|---|---|
| 12345678 | plant | valve |
| 12345679 | plant | valve |
| 12345680 | plant | pipe |
| 12345681 | plant | pipe |
| 12345682 | plant | <NULL> |

The second table is called `PVS_QUERY`. This table contains the following fields:

- `QUERY_TYPE` — The type of data to be returned (for example, properties, construction, scheduling)
- `OBJECT_TYPE` — The specific type of an object (for example, valve, pipe, steel). This field can be `NULL`. A `NULL` object type indicates that it is not needed to be used in searching for the query statement.
- `QUERY_NAME` — The name of the query statement used to locate the query text in the query statement table.

Below is an example of this table:

**Table A-2 Sample PVS_QUERY Table**

| QUERY_TYPE | OBJECT_TYPE | QUERY_NAME |
|---|---|---|
| construction | <NULL> | construction |
| properties | <NULL> | properties |
| construction | pipe | construction |
| properties | pipe | properties |
| construction | valve | construction |
| properties | valve | properties |

The third table is called `PVS_QUERY_MODE`. This table contains the following fields:

- `QUERY_TYPE` — The type of data.
- `QUERY_MODE` — The fieldname used to retrieve the appropriate query statement. It can have values of either `OBJECT_ID`, `FILE_NAME` or `NULL`. A `NULL` value indicates that both `OBJECT_ID` and `FILE_NAME` are used.

This table is optional. If it does not exist, the query mode is considered to be NULL.

Below is an example of this table:

**Table A-3 Sample PVS_QUERY_MODE Table**

| QUERY_TYPE | QUERY_MODE |
|---|---|
| construction | <NULL> |
| maintenance | OBJECT_ID |
| properties | OBJECT_ID |
| purchasing | FILE_NAME |

The fourth table is the PVS_SQL_STMT table. This table contains the following fields:

- QUERY_NAME — Query statement name used in the querying.
- SEQ_NUM — The number used in ordering lines that constitute a complete query. This field is incriminated when a query statement exceeds the maximum length of the QUERY_STMT field.
- QUERY_STMT — Query text.

Below is an example of this table:

**Table A-4 Sample PVS_SQL_STMT Table**

| QUERY_NAME | SEQ_NUM | QUERY_STMT |
|---|---|---|
| construction | 1 | select START, END from CONSTRUCTION, PVS_SUB_VAL where OBJECT_ID = @OBJECT_ID and CONSTRUCTION.LINE_NUM = PVS_SUB_VAL.LINE_NUM |
| construction | 2 | |
| construction | 3 | |
| construction | 4 | |
| purchasing | 1 | select PRICE, UNIT from PURCHASING where OBJECT_ID = @OBJECT_ID |
| purchasing | 2 | |
| tag | 1 | select TAG_TEXT from TAGS where OBJECT_ID = @OBJECT_ID |
| tag | 2 | |

When an object is digitized, Visualizer automatically registers three dynamic substitution variables: OBJECT_ID, FILE_NAME and QUERY_TYPE. The values for

OBJECT_ID and FILE_NAME are obtained from the digitized object and the value for QUERY_TYPE is obtained from the currently selected QUERY_TYPE in the Query Types dialog box. These variables are used by the query statement to obtain the verification information.

You can also use additional tables to provide substitution values for the queries. These tables are optional and are used if additional substitution values are needed.

Below is an example of this type of table:

| OBJECT_ID | FILE_NAME | LINE_NUM | VENDOR | PART_NUM |
|---|---|---|---|---|
| 12345678 | plant | LN-1000 | Tube-Turn | PN-1000 |
| 12345679 | plant | LN-1000 | Tube-Turn | PN-1001 |
| 12345680 | plant | LN-2000 | Crane | PN-2000 |
| 12345681 | plant | LN-2000 | Crane | PN-2000 |
| 12345682 | plant | LN-3000 | Crane | PN-2001 |

Table A-4, “Sample PVS_SQL_STMT Table” on page A-13 shows an example of how this table is used in a query statement in the construction query of the PVS_SQL_STMT table.

## Verification Procedures

Visualizer obtains verification information by performing the following steps:

1. The currently selected QUERY_TYPE from the Visualizer session is used with the PVS_QUERY table to locate the query mode. To do this, the following query statement is used:

```
select
      QUERY_MODE from PVS_QUERY_MODE
where
      QUERY_TYPE = @QUERY_TYPE
```

2. With the query mode returned from the above query, PVS_OBJECT and PVS_QUERY tables are joined to locate the query name. The OBJECT_ID and FILE_NAME fields of PVS_OBJECT table are matched with the information obtained from the digitized object, and the value of the QUERY_TYPE field of the PVS_QUERY table is matched with the query type obtained from Visualizer. If the query mode is OBJECT_ID, only the OBJECT_ID is used to locate the query name. To do this, the following query statement is used:

```
select
      QUERY_NAME from PVS_OBJECT, PVS_QUERY
where
```

```
      OBJECT_ID = @OBJECT_ID and
      QUERY_TYPE = @QUERY_TYPE and
      (PVS_QUERY.OBJECT_TYPE = PVS_OBJECT.OBJECT_TYPE or
            (PVS_QUERY.OBJECT_TYPE is NULL and
             PVS_OBJECT.OBJECT_TYPE is NULL))
```

3. If the query mode is `FILE_NAME`, only the `FILE_NAME` is used to locate the query name. To do this, the following query statement is used:

```
select
      QUERY_NAME from PVS_OBJECT, PVS_QUERY
where
      FILE_NAME = @FILE_NAME and
      QUERY_TYPE = @QUERY_TYPE and
      (PVS_QUERY.OBJECT_TYPE = PVS_OBJECT.OBJECT_TYPE or
       (PVS_QUERY.OBJECT_TYPE is NULL and
        PVS_OBJECT.OBJECT_TYPE is NULL))
```

4. If the query mode is `NULL`, both `OBJECT_ID` and `FILE_NAME` are used to locate the query name. To do this, the following query statement is used:

```
select
      QUERY_NAME from PVS_OBJECT, PVS_QUERY
where
      OBJECT_ID = @OBJECT_ID and
      FILE_NAME = @FILE_NAME and
      QUERY_TYPE = @QUERY_TYPE and
      (PVS_QUERY.OBJECT_TYPE = PVS_OBJECT.OBJECT_TYPE or
       (PVS_QUERY.OBJECT_TYPE is NULL and
        PVS_OBJECT.OBJECT_TYPE is NULL))
```

5. The query name returned is used to locate the text of the query statement in the `PVS_SQL_STMT` table. To do this, the following query statement is used:

```
select
      QUERY_STMT from PVS_SQL_STMT
where
      QUERY_NAME = :QUERY_NAME
      order by SEQ_NUM asc
```

For example, if you digitize an object with `OBJECT_ID` equal to `12345680` and the `FILE_NAME` is `plant`, and the query type selected in Visualizer is `purchasing`, with the preceding example tables, the following query statement is retrieved:

```
select
      PRICE, UNIT from PURCHASING
where
      OBJECT_ID = @OBJECT_ID
```

A substitution variable `OBJECT_ID` with a value of `12345680` is registered. This query statement is submitted and the values returned are displayed in the Database Information section of the Object Verification dialog box.

# Automatic Tag Tool

The Tag tool labels the digitized object in Visualizer with the object data retrieved from the connecting DBMS.

Automatic tagging is done using the same mechanism as the Verify command with one exception: the query type is not obtained from the Query Type list; it is always `TAG`. The returned value is used as the tag data for the digitized object.

# Environment Setup

Visualizer SQL Interface requires some setup on the hosts running Oracle and hosts running Visualizer. Before using Visualizer SQL Interface, perform the following steps:

## On all Hosts

1. Register the DBMS network daemon program ID by adding `dbmsnetd 200007` to all the `/etc/rpc` files. You can also accomplish this with the following command:

   ```
   setenv dbmsnetd 200007
   ```

2. Set the Oracle RDBMS system identifier with the following command:

   ```
   setenv ORACLE_SID SID_name
   ```

   The `ORACLE_SID` for the host running ORACLE and the hosts running Visualizer must be the same.

## On Hosts Running Oracle

Execute the DBMS daemon. Daemons for supported platforms are distributed with Visualizer and located in the `common/sqlbin` directory.

## On Hosts Running Visualizer

Specify the DBMS host with the following command:

```
setenv DBMS_HOST name_of_host_running_Oracle
```

# DBMS Demo

This section explains how to use the DBMS demo, which is designed to help you understand the SQL interface to Visualizer.

The following executable files populate the database with test data, and reside in the `${VIS_HOME}/bin` directory:

- `parsedat` — Reads an ASCII database definition file and populates the data in the connecting DBMS database.
- `parsedb` — Reads the data file and populates the data in the connecting DBMS database.

The following files are used by the preceding executable files, and reside in the `${VIS_HOME}/common/sqldemo` subdirectory:

- `popdb` — A `csh` script that accepts an Oracle usercode, password, and database definition file, which calls the `parsedat` and `parsedb` executable files.
- `demo.db` — Database for demonstrating the SQL Interface.
- `demo.def` — Database definition file for demonstrating the SQL Interface.
- `object.dat` — Data file for the object table `pvs_object`.
- `purchasing.dat` — Data file for the purchasing table `pvs_purchasing`.
- `query.dat` — Data file for the query table `pvs_query`.
- `sql_stmt.dat` — Data file for the statement table `pvs_sql_stmt`.
- `nosubvar.sql` — File containing an SQL statement that does not use substitution variables.
- `subvar.sql` — File containing an SQL statement that uses substitution variables.
- `dbms.gaf` — A `.gaf` file containing definitions for query types and selection keys.

## Before Using SQL Interface

Before using the SQL Interface, make sure Oracle is running on a node on the network.

### On All Hosts

Register the DBMS network daemon program ID by adding `dbmsnetd 200007` to all the `/ect/rpc` files. Alternately, you can set an environment variable, as follows:

```
setenv dbmsnetd 200007
```

### On Hosts Running Oracle

1. Set Oracle SID by entering the following command:

   `setenv ORACLE_SID` SID_name

2. Execute DBMS daemon. Daemons for supported platforms are distributed with the product software and are located in the following directory:

   `${VIS_HOME}/common/sqlbin.mach_os`

### On Hosts Running Visualizer

1. Set Oracle SID by entering the following command:

   `setenv ORACLE_SID` SID_name

2. Specify the DBMS host as follows:

   `setenv DBMS_HOST` Oracle_hostname

3. Change to the following demonstration directory:

   `${VIS_HOME}/common/sqldemo`

4. Populate the database by entering the following command:

   `popdb` Oracle_username Oracle_password

5. Enter `demo.def` for the definition file.

These steps create two tables, `PLX99` and `PLFITTING`, containing a total of eight items. This is accomplished by the `parsedb` program. For more information on this program, refer to the `parsedb.man` page.

Tables for object verification are also created. The valid object types are construction, properties, and purchasing. This is accomplished by the `parsedat` program. For information on this program refer to the `parsedat.man` page located in `${VIS_HOME}/common/sqldemo`.

## Testing the Interface, Query, and Verification

The procedures for testing the Visualizer interface, Query database and related functions, and verification are described in the following sections.

## Testing the Visualizer Interface

After creating the necessary tables, you are ready to test out the interface as follows:

1. Execute Visualizer by entering the following:

   ```
   ${VIS_HOME}/scripts/visualizer
   ```

2. Open the metafile by entering or selecting the following:

   ```
   ${VIS_HOME}/common/data/plant.mf
   ```

3. Connect to DBMS by selecting Database > Connect > DBMS from the Visualizer menu bar. In the DBMS Connect window, enter the User ID and password, then click Connect.
4. Add construction, purchasing, and properties query types by selecting Database > Query Types and specifying the information in the New Query Types window. Note that you can specify only one type at a time.
5. Select Database > Selection Keys, then change the selection key for the file name field from `FILE_NAME` to `MODEL`.

## Testing the Query Database

To test the Query Database and the related functions, select Database > Query Database. In the Query Database window, click Select Query from File. In the Select Query File window displayed next, select or specify `subvar.sql` or `nosubvar.sql`. These files contain simple queries, and reside in the `${VIS_HOME}/common/sqldemo` directory.

The `subvar.sql` file shows the use of substitution variables. For the query that uses a substitution variable, enter either `'PCV101'` or `'PCV102'` in the substitution value text field then press `Return`. Note that these values must be enclosed in single quotes. Click Submit to submit the query.

Each query you submit displays the Query Database Data window, which contains the result of the query. If you click on the record data to select it, you can then select Zoom, Highlight, or Add to Selection List. To return to the original view, select View > Primary View > Show from the Visualizer window.

## Testing Verification

To test the verification function, select Database > Query Database. In the Query Database window, click Select Query from File. In the Select Query File window displayed next, specify `nosubvar.sql` then submit the entry.

In the Query Database Data window, click on the record data then select Zoom and Highlight. Next, select the Verification Tool from the Visualizer tool palette, and use it to select the valve that is highlighted. The Object Verification Window should now display information about what was selected, and all the database information that has been entered is also stored.

Please note: Database information remains blank if the query types were not added in a previous build, or if no information has been entered on the selected item.

Select the Verification Tool and use it to select an object.

In order to see all the data populated by `parsedb` and `parsedat` you can submit the following queries one-at-a-time in the Query Database window:

```
SELECT * FROM PLX99
SELECT * FROM PLFITTING
SELECT * FROM PVS_OBJECT
SELECT * FROM PVS_QUERY
SELECT * FROM PVS_SQL_STMT
```

Appendix B

# EPD.Visualizer Textual Graphic File

This appendix provides information about the format and structure of the EPD.Visualizer textual graphic file. Examples of actual graphics information are provided to help you create or edit a textual graphic file.

The following topics are discussed in this appendix:

- Textual Graphic File Contents
- Textual Graphic File Format
- Control Options
- DBMS Control Options
- Extents Definitions
- Color Definitions
- Light Definitions
- Material Definitions
- Path Definitions
- GPath Definitions
- Object Definitions
- Gpath Multiple Sectioning

# Textual Graphic File Contents

The textual graphic file is usually generated by your host CAD system, but you can also create a textual graphic file with a standard text editor.

A textual graphic file can contain the following sections:

- Control Options
- Extents Definitions
- Color Definitions
- Light Definitions
- Material Definitions
- Path Definitions
- Object Definitions

EPD.Visualizer does not limit the size of the graphic file. However, no single line in the file should exceed 256 characters because lines longer than 256 characters might not be processed.

# Textual Graphic File Format

If you want to create a new graphic file or edit an existing one, you need to understand how to format and organize graphics information.

All EPD.Visualizer textual graphic files begin with the key word `#GRAPHICS_FILE` and end with the key word `#END`. You can enter spaces or tabs within the file for readability. You can also add comments anywhere in a graphic file by prefacing the comment with an exclamation point (`!`).

Between `#GRAPHICS_FILE` and `#END`, graphics information is organized into hierarchical sections. Each section within the file must begin with a key word section name and end with the key word `#END`. The Control Options section must be the first section in the file. If the Extents Definition section is included, it must appear before the sections defining the geometry. The other sections can be in any order.

The following example shows the basic structure of the textual graphic file with variations in the hierarchical form of the file. Specific details are described in the sections following the example.

```
#GRAPHICS_FILE

      #section_name

      #END

      #section_name
            #section_name

            #END
      #END

      #section_name
            #section_name
                  #section_name

                  #END
                  #section_name

                  #END
            #END
            #section_name

            #END
      #END
      .
      .
      .
#END
```

# Control Options

Control options are defined in a section identified by the key word `#CONTROL_OPTIONS`. The control options section is required for all textual graphic files and must be the first section in the file. Currently, the only information allowed in this section is the geometry unit designation.

EPD.Visualizer uses this unit designation to properly convert geometry in your file to a uniform unit type when more than one graphic file is added. Valid unit types include the following:

- mils
- inches
- feet
- yards
- miles
- millimeters
- centimeters
- decimeters
- meters
- kilometers

The following example shows the format for the control options section:

```
#CONTROL_OPTIONS
  unit = inches
#END
```

# DBMS Control Options

DBMS related options are defined in an optional section identified by the key word `#DBMS`. This section is used to define query types and `object_id` and `file_name` selection keys used by the EPD.Visualizer SQL Interface. Query types are the types of data to be retrieved by the Object Verification tool. Selection keys are the field names EPD.Visualizer uses to select items returned from the connecting DBMS.

The following example shows the format for defining query types and selection keys.

```
#DBMS
  query_types = (construction, purchasing)
  object_id = OBJECT_ID
  file_name = MODEL
#END
```

# Extents Definitions

Extents are defined in an optional section identified by the key word `#EXTENT`. If this section is provided, it must appear before any sections where geometry is defined.

The extents section contains the definition of the three-dimensional boundaries, or extents, of all geometry contained in a graphic file. Typically, this section is generated by your host CAD system. However, if you are creating a textual graphic file and omit this section, EPD.Visualizer automatically computes the 3D extents when the graphic file is added during a EPD.Visualizer session.

The following example shows the format for defining 3D extents using actual decimal values:

```
#EXTENTS
  xmin =-10.0
  ymin =-10.0
  zmin =-10.0
  xmax =10.0
  ymax =10.0
  zmax =10.0
#END
```

# Color Definitions

The color definition section is identified by the key word `#COLORS`. This section is used to define colors that you can use when defining the color of items in EPD.Visualizer.

Each color defined in the colors section must be given a unique name. Color intensity is defined using three floating point values that represent red, green, and blue intensities (RGB), in that order. Valid RGB values range from 0.0 to 1.0 where 1.0 represents the highest intensity allowable. You can define any number of colors within the limits of the color table on your workstation.

The following example shows the format for defining colors:

```
#COLORS
red   =(1.00.00.0)
green =(0.01.00.0)
blue  =(0.00.01.0)
cyan  =(0.01.01.0)
gray  =(0.50.50.5)
black =(0.00.0.0)
white =(1.01.01.0)
#END
```

# Light Definitions

There are three types of lights — directional, point, and spot. Directional lights are defined in a section identified by the key words `#DIRECTIONAL_LIGHT`. Point lights are defined in a section identified by the key words `#POINT_LIGHT`. Spot lights are defined in a section identified by the key words `#SPOT_LIGHT`. You can define any number of lights in a graphic file. Each light must have a unique name.

The following tables show the key words, allowable values and settings, and defaults for the light definition attributes. Certain key words are optional; others are required and indicated with (R). If you omit an allowed value or setting, the default is used.

Table B-1 shows the directional light definition attributes.

**Table B-1 Directional Light Definition Attributes**

| Key Words | Allowed Values/Settings | Defaults |
|---|---|---|
| name (R) | 1-32 characters | none |
| color | color name or RGB values | white |
| switch | on or off | on |
| direction (R) | XYZ coordinates in decimal values between 0 and 1 | none |

The following example shows the format for defining directional lights:

```
#DIRECTIONAL_LIGHT
 name = dir_light1
 color = white
 switch = on
 direction = 0.3 0.2 1.0
#END
```

Table B-2 shows the point light definition attributes.

**Table B-2 Point Light Definition Attributes**

| Key Words | Allowed Values/Settings | Defaults |
|---|---|---|
| name (R) | 1-32 characters | none |
| color | color name or RGB values | white |
| switch | on or off | on |
| position (R) | xyz coordinates in decimal values | none |

The following example shows the format for defining point lights:

```
#POINT_LIGHT
 name = point_light1
 color = white
 switch = on
 position = 2.0 0.0 2.0
#END
```

Table B-3 shows the spot light definition attributes.

**Table B-3 Spot Light Definition Attributes**

| Key Words | Allowed Values/Settings | Defaults |
|---|---|---|
| name (R) | 1-32 characters | none |
| color | color name or RGB values | white |
| switch | on or off | on |
| position (R) | XYZ coordinates in decimal values | none |
| target (R) | XYZ coordinates in decimal values | none |
| illumination_angle (R) | 0-180 degrees | 30 degrees |
| edge_sharpness (R) | 0-100 | 80 percent |
| symbol_size | decimal value | distance between position and target |

The following example shows the format for defining spot lights:

```
#SPOT_LIGHT
 name = spot_light1
 color = white
 switch = on
 position = 2.0 0.0 2.0
 target = 3.0 0.0 4.0
 illumination_angle = 45
 edge_sharpness = 90
 symbol_size = 3
#END
```

# Material Definitions

Materials are defined in a section identified by the key word `#MATERIAL`. Material definitions can be assigned to objects in EPD.Visualizer for use in ray tracing. You can define any number of materials in a graphics file. Each material must have a unique name.

You can either define a material with or without color. For example, a material named copper would have a base color while a material named `shiny_paint` would be colorless.

Table B-4 shows the key words, allowable values and settings, and defaults for the material definition attributes. Certain key words are optional; others are required and indicated with (R). If you omit an allowed value or setting, the default is used.

**Table B-4 Material Definition Attributes**

| Key Words | Allowed Values/Settings | Defaults |
|---|---|---|
| name (R) | 1-32 characters | none |
| base_color | color name or RGB values | colorless |
| specular_hue | base_color_shades gray_shades | base_color_shades |
| reflectance | integers 0 — 10<br>0 = none<br>10 = maximum | 0 |
| transparency | integers 0 — 10<br>0 = opaque<br>10 = clear | 0 |
| specular_intensity | integers 0 — 10<br>0 = dull<br>10 = shiny | o |
| specular_glare (R) | integers 0 — 10<br>0 = low<br>10 = high | 5 |
| symbol_size | decimal value | distance between position and target |

The following example a graphics file with material definitions:

```
#MATERIAL
name = polished_copper
base_color = (0.79 0.45 0.15)
specular_hue = base_color_shades
reflectance = 5
transparency = 0
specular_intensity = 5
specular_glare = 10
#END

#MATERIAL
name = shiny_paint
specular_hue = gray_shades
reflectance = 1
transparency = 0
specular_intensity = 7
specular_glare = 7
#END

#MATERIAL
name = green_glossy_paint
base_color = green
specular_hue = gray_shades
reflectance = 1
transparency = 0
specular_intensity = 10
specular_glare = 10
#END

#MATERIAL
name = smoked_glass
base_color = (0.4 0.4 0.4)
specular_hue = gray_shades
reflectance = 4
transparency = 5
specular_intensity = 10
specular_glare = 10
#END
```

# Path Definitions

Paths are defined in a section identified by the key word `#PATH`. Paths are used by item and camera traversal. You can define any number of path sections in a graphic file. Each path must have a unique name and an assigned color.

The path points subsection is identified by the key word `#POINTS`. Each point is defined using three floating point values that represent an xyz location. The first point in the path points definition specifies the head of the path. You can define any number of path points.

The following example shows the format for defining paths:

```
#PATH
name = fire_route_1
color = red
#POINTS
     0.0-5.05.0
     0.00.0 5.0
    30.00.0 5.0
#END
#END

#PATH
name = pathway_1
color = blue
#POINTS
    90.0-10.00.0
    90.0-10.0-5.0
    20.0-10.0-5.0
    20.0-10.00.0
    20.020.00.0
#END
#END
```

# GPath Definitions

Gpaths are defined in a section identified by the keyword #GPATH. Sectioning planes and objects can traverse Gpaths, or translate and rotate by moving to a Gpath reference. You can define any number of Gpath sections in a graphic file.

The Gpath points (references) subsection is identified by the keyword `#REFPOINTS`. Each reference is defined by the following:

- three floating point values that represent an XYZ location
- three floating point values indicating the rotation angles around the global axis in degrees: AX, AY, AZ
- the name of the reference point (optional).

  A default name is automatically built by adding an order number to the string defined in the PVS_UNNAMED_REF environment variable. If the environment variable is not defined, the default string is `rf`. There is no theoretical limit to the length of the name but it is reasonable to that a maximum of 15 characters defines a name.

The first reference in the Gpath points definition specifies the head of the path. You can define a maximum of 500 references in a Gpath.

The following example shows the definition of a Gpath:

```
#GPATH
name = newpath
color = yellow
      #REFPOINTS
      195.0       851.0 593.0 178.0 0.0   179.9
Start
      -240.0      815.0 587.0 73.0  0.0   -171.5      r2
      -406.0      807.0 516.0 99.6  90.0  9.4         r3
      389.0       849.0 329.0 99.9  90.0  176.2       r4
      -15.0       -58.0 328.0 90.0  0.0   180.0       r5
      -4.0        -43.0 100.0 86.8  10.0  90.0        End
      #END
#END
      - 4.0       -43.0 100.0 86.8  10.0  90.0  End
```

# Object Definitions

Objects are defined in a section identified by the key word `#OBJ`. Typically, the object definition section is generated by your host CAD system. However, if you are creating a textual graphic file, you can define geometric objects in this section.

Each object in a textual graphic file must be assigned a unique object identifier so that EPD.Visualizer can easily differentiate between the geometric objects. You can also assign additional optional attributes such as discipline, group, and color.

The discipline attribute specifies that an object belongs to a particular engineering discipline, such as electrical, hydraulics, piping, structural, or drive train. The group attribute assigns an object to an EPD.Visualizer group, which is often equivalent to a level, layer, or class on your CAD system. EPD.Visualizer has many features that allow you to perform on disciplines and groups to make editing convenient and easy-to-use.

The following examples show the format for defining object attributes (geometries definitions are not shown):

```
#OBJ
object_id = -2124343233
discipline = structural
group = 100
color = aquamarine
.
.
.
#END


#OBJ
object_id = id_9823_t
discipline = drive_train
group = transmission
color = metallic_gray
.
.
.
#END
```

From a geometric viewpoint, all objects are defined using polygons and polylines. Polygons are defined in accordance with the right-hand rule. This rule specifies that polygon points are ordered in a counterclockwise direction with respect to the outer or visible side of the polygon face.

Polygon points that define a face must be coplanar, which means that the points must define a flat plane.

Please note: Polygons that are not coplanar might display erratically.

Polygons are specified in two formats — flat or shell polygons. The shell format defines polygons as a network of interconnected polygons that share common points and result in a unit typically called a polyhedron. Shells can be smooth shaded by EPD.Visualizer, while flat polygons cannot. Polylines are connected sequence of lines.

# Flat Polygon Definition

A flat polygon is defined in a subsection of object definition and identified by the key word `#FP`. You can create any number of flat polygon subsections.

A polygon point is defined using three floating point values that represent an xyz location. Flat polygons require a minimum of three points to define a plane.

The following example shows the format for defining flat polygon points:

```
#FP
10.0  0.00.0
20.0  0.00.0
20.0  5.00.0
10.0  5.00.0
#END
```

# Shell Definition

A shell polyhedron is defined in a subsection of object definition and identified by the key word `#SP`. You can have any number of shell sections within an object section. Shells are defined by a network of connected faces that share common points. Shell points and faces are defined in separate subsections.

## Shell Points Definition

Shell points are defined in a subsection of shell definition and identified by the key word `#POINTS`. Points can be specified in any order. The points are defined in three floating point values representing an xyz location.

The following example shows the format for defining shell points:

```
#POINTS
 70.5 -20.90.5
 60.0 -10.0-5.0
 90.0 -15.0-5.5
 30.2 -10.10.0
 15.0 20.02.0
 10.7 30.22.4
 10.0 0.00.0
#END
```

### Shell Faces Definition

Shell faces are defined in a subsection of shell definition and identified by the key word `#FDS`. The shell faces definition subsection must follow the shell points subsection.

Each shell face that you define must have at least three points to define a valid face. A shell face is defined by specifying the number of points that define a face, followed by a list of integers that index into the point list. This list of integers identifies the points to use when defining a face. The first point in the point section is indexed at position 0.

A hole in a shell is defined in the same manner as a regular face, except that the integer defining the number of points for each face is negative instead of positive. The negative number instructs EPD.Visualizer to subtract the face from the shell definition. This results in the display of a hole.

The following example shows the format for defining shell faces and holes:

```
#FDS
5   0  12  3 6
3   2  34
5   3  02  4 5
-4  2  53  0
#END
```

## Polyline Definition

A polyline is defined in a subsection of object definition identified by the key word `#PL`. You can create any number of polyline subsections.

A poly point is defined with three floating point values that represent an xyz location. Polylines require a minimum of two points to define a line.

The following example shows the format for defining flat polygon points:

```
#PL
 20.0  0.00.0
 30.0  0.00.0
 30.0  5.00.0
 20.0  5.00.0
#END
```

# Object Definition Example

The following example shows a complete object definition section using flat polygons and shells:

```
#OBJ
object_id = id_9823_t
discipline = drive_train
group = transmission
color = metallic_gray
#FP
   10.0  0.0  5.0
   10.0  0.0  5.0
   20.0  7.0  5.0
   -3.0  5.0  .0
#END
#SP
   #POINTS
      70.5  -20.9  0.5
      60.0  -10.0 -5.0
      90.0  -15.0 -5.5
      30.2  -10.1  0.0
      15.0   20.0  2.0
      10.7   30.2  2.4
   #END
   #FDS
      4  0  1  2  3
      3  2  3  4
      5  3  0  2  4  5
     -4  2  5  3  0
   #END
#END
#PL
   20.0  0.0  0.0
   30.0  0.0  0.0
   30.0  5.0  0.0
#END
#END
```

# Gpath Multiple Sectioning

When you save the result of Gpath multiple sectioning in an EPD.Visualizer textual graphic file, the file has the following format:

```
#GRAPHICS_FILE
#CONTROL_OPTIONS
unit = ...        The unit is defined from the "Options|
                  Preferences" menu (Working Unit)
#END
< Section generated by the 1st visible reference >
!
< Section generated by the 2nd visible reference >
!
..../....
< Section generated by the nth visible reference >
#END
```

The section generated by a reference can be described by the following format:

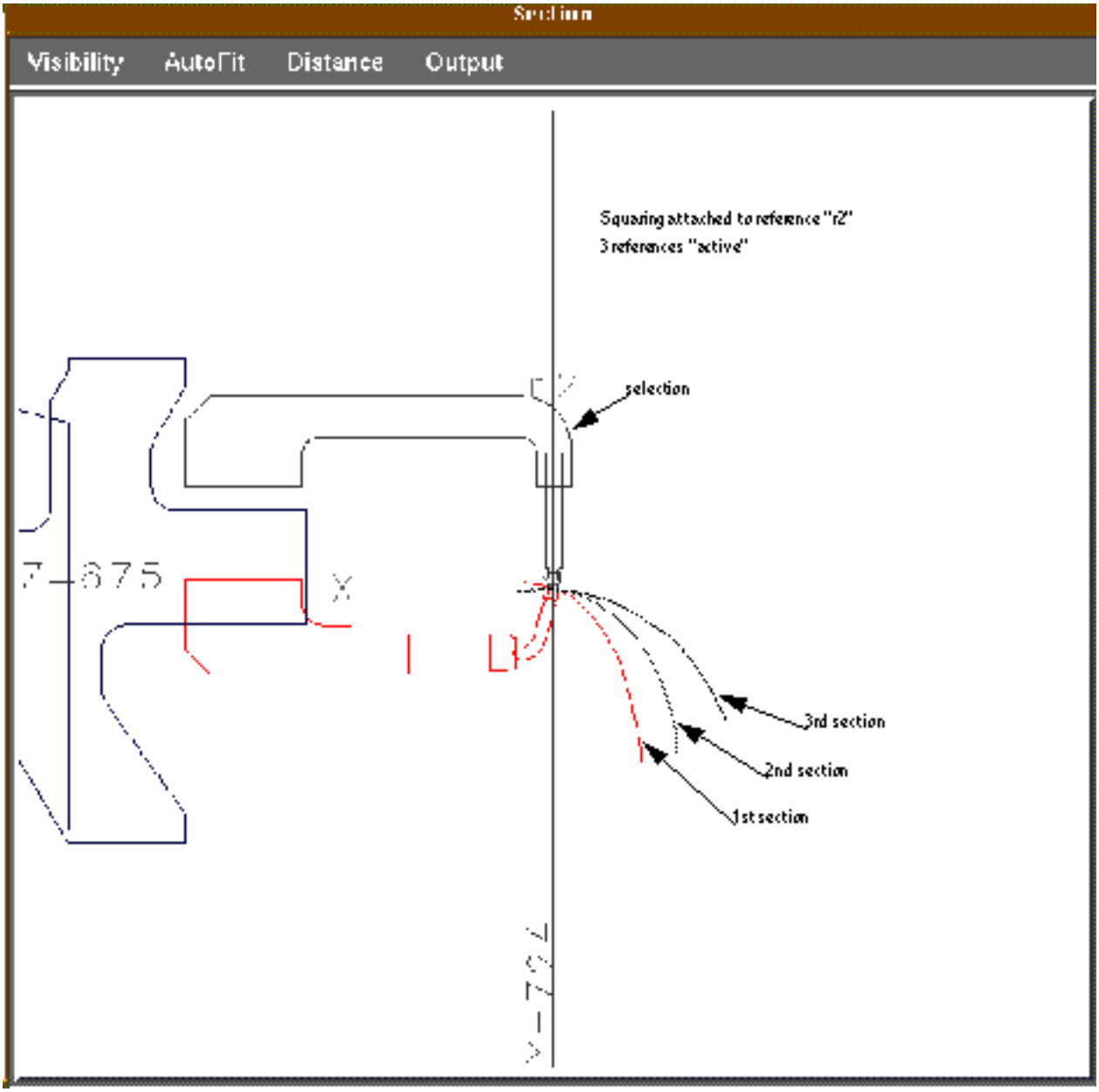
```
#COLORS
Definition of the color attached to the reference
#END
< Desc. of section of graphic file 1 >
< Desc. of section of graphic file 2 >
..../....
< Desc. of section of graphic file n >
```

The section of a graphic file is an object made of one or several polylines. The name of the sectioned object is `object_id`. The name of the color attached to the reference is `color`. The `discipline` is always `gpathsection`. The following example shows the format of the generated file:

```
#GRAPHICS_FILE
#CONTROL_OPTIONS
      unit = millimeters
#END
#COLORS
      bisque = (1.000000 0.894121 0.768627)
#END
#OBJ
      object_id = "referrage_1$collecteur-auvent_1"
      discipline = gpathsection
      group = "reference_1"
      color = bisque
      #PL
            -419.457916 0.000000 -123.002235
      .../...
            -319.848602 0.000000 34.280548
      #END
#END
```

```
#OBJ
      object_id = "breferrage_1$doublure-aile-av_1"
      discipline = gpathsection
      group = "reference_1"
      color = bisque
      #PL
      -311.016663 0.000000 -602.096375
      .../...
      -202.332581 0.000000 -539.660095
      #END
#END
!
#COLORS
      ivory = (1.000000 1.000000 0.941176)
#END
#OBJ
      object_id = "referrage_1$collecteur-auvent_1"
      discipline = gpathsection
      group = "reference_2"
      color = ivory
      #PL
      -183.442596 0.000000 280.628540
      .../...
      -318.448303 0.000000 -20.776367
      #END
      #PL
      -183.404480 0.000000 280.794495
      .../...
      -207.700714 0.000000 308.379639
      #END
#END
#OBJ
      object_id = "breferrage_1$doublure-aile-av_1"
      discipline = gpathsection
      group = "reference_2"
      color = ivory
      #PL
      -231.227173 0.000000 -556.665161
      .../...
      -429.578644 0.000000 -566.814026
      #END
#END
#END
```

The preceding file describes the following Section window.




The selected object that traversed the Gpath, in this case a spot-welding tool, appears only once in the section window, and it is not described in the saved textual graphic file. For other objects, each active reference may generate a section in the section window, and each section has a description in the textual graphic file.

Appendix C

# EPD.Visualizer Binary Graphic File

This appendix provides information about the format and structure of the EPD.Visualizer binary graphic file. Examples of actual graphics information are provided to help you create or edit a binary graphic file.

Binary graphic files are designed to contain the same information as the textual graphic file, with the exception of light, material, and path definitions. (The light, material, and path definition sections are not available for binary graphic files.) The information represented in binary format is more compact and can be processed at a much faster rate. However, binary graphic files are system dependent and the binary data generated must be compatible with or convertible by EPD.Visualizer.

The binary graphic file is a contiguous block of binary data. Line breaks, blank space(s), tabs, and field separators ( | ) have been inserted in the examples for clarification.

The following topics are discussed in this appendix:

- Binary Graphic File Format
- Control Options
- Extents Definition
- Color Definition
- Object Definition

# Binary Graphic File Format

All EPD.Visualizer binary graphic files must begin with a file identifier, which is an unsigned 1-byte character indicating the system on which it was generated. The following are the currently accepted unsigned 1-byte characters:

- A value of 255 (hex FF) indicates that the binary graphic file was generated on a DECstation where the byte and bit order are backward
- A value of 254 (hex FE) indicates that the binary graphic file was generated on a Sun or HP workstation, or another type of processor, where the byte order is forward and the bit order is backward

Integers and floating point numbers in the binary graphic file must be compatible with the format of the EPD.Visualizer host machine. EPD.Visualizer provides conversion between DECstation, Sun, or HP binary graphic files. For example, if EPD.Visualizer is running on the Sun, binary graphic files in DECstation format can be read, and binary graphic files in Sun format can be read on the DECstation.

Following the file identifier are the control options, extents definition, color definition, and object definition sections. Refer to Appendix B, “EPD.Visualizer Textual Graphic File” for more information on these sections.

Each section starts with a 2-byte integer that serves as the section identification number, followed by the contents specific to that section. Table C-1 shows the section identification number values:

**Table C-1 Section Identification Number Values**

| Sections | Value |
|---|---|
| `extents_section_number` | 0 |
| `color_section_number` | 1 |
| `object_section_number` | 2 |
| `control_options_section_number` | 3 |

# Control Options

The control options section is required for all binary graphic files and must be the first section in the file. The following is the format for the control options section:

```
control_options_section_number
number_of_options
control_options_specifications
```

In this format, control_options_section_number is a 2-byte integer equal to 3, and number of options is a 4-byte integer specifying the number of options.

Following the control_options_section_number is the control_options_specifications. Each specification is preceded by an option specification number. Currently, the only information allowed in this section is the geometry-unit specification; therefore, the number_of_options is equal to 1.

The format for the geometry units for control_options_specifications is the following:

```
unit_attribute_number
unit_type
```

In this format, unit attribute number is a 2-byte integer equal to 6. Following the unit_attribute_number is the unit_type. Table C-2 shows the values corresponding to the unit types:

**Table C-2 Unit Type Values**

| Unit Type | Value |
|---|---|
| mils | 1 |
| inches | 2 |
| feet | 3 |
| yards | 4 |
| miles | 5 |
| millimeters | 6 |
| centimeters | 7 |
| decimeters | 8 |
| meters | 9 |
| kilometers | 10 |

The following examples show a textual control options section and a corresponding binary control options section in hexadecimal notation:

Textual Control Options

```
#CONTROL_OPTIONS
      unit = inches
#END
```

Binary Control Options

```
00030000000100060002 —> #CONTROL_OPTIONS(3) 1 unit(6) inches(2)
|  ||      ||  ||  |
```

# Extents Definition

The extents definition section is optional. If provided, this section must appear before any sections where geometry is defined.

The following is the format for the extents definition section:

```
extents_section_number
xmin ymin zmin xmax ymax zmax
```

In this format, extents_section_number is a 2-byte integer equal to 0, and xmin, ymin, zmin, xmax, ymax, and zmax are 4-byte single-precision floating-point numbers defining the extents of the geometry.

The following examples show a textual extents definition section and a corresponding binary extents definition section in hexadecimal notation:

Textual Extents Definition

```
#EXTENTS
      xmin = -10.0
      ymin = -10.0
      zmin = -10.0
      xmax =  10.0
      ymax =  10.0
      zmax =  10.0
#END
```

Binary Extents Definition

```
0000C0240000C0240000C0240000—> #EXTENTS(0)-10.0 -10.0 -10.0
|  ||      ||      ||      |
      40024000400240004002400—> 10.0  10.0  10.0
      |      ||      ||      |
```

# Color Definition

The color definition section is optional. This section is used to define colors that you can use when defining the color of geometric items. You can define any number of colors within the limits of the color table on your workstation.

The following is the format for the color definition section:

color_section_number
number_of_colors
color_descriptions

In this format, color_section_number is a 2-byte integer equal to 1 and number of colors is a 4-byte integer specifying the number of colors. The following is the format for color_descriptions:

color_name_length
color_name
color_intensities

In this format, color_name_length is a 4-byte integer and color name is an ASCII string with length specified by the color_name_length value.

The color_intensities is defined with three 4-byte single-precision floating-point numbers that represent red, green, and blue intensities (RGB), in that order. Valid RGB values range from 0.0 to 1.0 where 1.0 represents the highest intensity allowable.

The following examples show a textual color definition section and a corresponding binary color definition section in hexadecimal notation:

Textual Color Definition

```
#COLORS
      red   = (1.0  0.0  0.0)
      green = (0.0  1.0  0.0)
      blue  = (0.0  0.0  1.0)
#END
```

## Binary Color Definition

```
000100000003 —> #COLORS(1) number of
|  ||      |colors(3)
000000037265643FFF0000000000000000000000—> name length(3) 'red'
|     ||    ||    ||    ||    |           1.0 0.0 0.0
00000005677265656E000000003FFF000000000000—> name length(5)
'green'
|     ||    ||    ||    ||    |           0.0 1.0 0.0
00000004626C7565000000000000000003FFF0000—> name length(4) 'blue'
|      ||      ||      ||      ||      |0.0 0.0 1.0
```

# Object Definition

The object definition section is used to define geometric objects for the graphic file. The following is the format for the each object definition section:

object_section_number
object_attributes
geometry_specifications

In this format, object_section_number is a 2-byte integer equal to 2. The following are the valid object_attributes:

discipline_attribute
group_name_attribute
object_id_attribute
color_attribute

The following table shows the values corresponding to the object attribute numbers.

**Table C-3 Object Attribute Numbers and their Values**

| Attribute Number | Value |
|---|---|
| discipline_attribute_number | 0 |
| group_name_attribute_number | 1 |
| object_id_attribute_number | 2 |
| color_attribute_number | 3 |

The following is the format for discipline_attribute:

discipline_attribute_number
attribute_length
attribute_value

The following is the format for group_name_attribute:

group_attribute_number
attribute_length
attribute_value

The following is the format for object_id_attribute:

object_id_attribute_number
attribute_length
attribute_value

The discipline_attribute_number, group_attribute_number, and object_id_attribute_number are 2-byte integers with a value equal to the value in Table C-2 on page C-3. The attribute_length is a 4-byte integer, and attribute_value is an ASCII string with the length specified by the attribute_length value.

The color_attribute is either a symbolic value in which a color name previously defined in the color definition section is used or a literal value in which the actual RGB values are specified.

The following is the format for the symbolic color attribute:

```
color_attribute_number
symbolic_color_type_number
attribute_length
attribute_value
```

In this format, the color_attribute_number is a 2-byte integer with a value equal to the value in Table C-3, "Object Attribute Numbers and their Values" on page C-8 and symbolic_color_type_number is a 2-byte integer with a value equal to the value in Table C-4, "Color Type Number and their Values" on page C-10, attribute_length is a 4-byte integer, and attribute_value is a ASCII string specifying a color name previously defined with length specified by the attribute length value.

The following is the format for the literal color attribute:

```
color_attribute_number
literal_color_type_number
red_value green_value blue_value
```

In this format, the color_attribute_number is a 2-byte integer with a value equal to the value in Table C-3, "Object Attribute Numbers and their Values" on page C-8 and the literal color attribute is a 2-byte integer with a value equal to the value in Table C-4, "Color Type Number and their Values" on page C-10, and red_value, green_value, and blue_value are 4-byte single-precision floating-point numbers that represent red, green, and blue values (RGB), in that order. Valid RGB values range from 0.0 to 1.0, where 1.0 represents the highest value allowable.

The following table shows the color type numbers and their corresponding values for color_attribute. These numbers are 2-byte integers.

**Table C-4 Color Type Number and their Values**

| Color Type Number | Value |
|---|---|
| symbolic_color_type_number | 2 |
| literal_color_type_number | 3 |

The following example shows a textual object definition section with a symbolic color attribute and a corresponding binary object definition section in hexadecimal notation:

Textual Object Definition

```
#OBJ
      object_id = -2124343233
      discipline = structural
      group = 100
      color = aquamarine
      .
      .
#END
```

Binary Object Definition

```
000200020000000B2D32313234333433323333—> #OBJ(2) object_id(2)
|  ||  ||      ||                     |11 '-2124343233'
      00000000000A737472756374757261 6C—> discipline(0) 10
      |  ||      ||                    |'structural'
      000100000003313030—> group(1) 3 '100'
      |  ||      ||    |
      000300020000000A617175616D6172696E65—> color(3) symbolic(2)
      |  ||  ||      ||                   |10 'aquamarine'
```

The following example shows a textual object definition section with a literal color attribute and a corresponding binary object definition section in hexadecimal notation:

Textual Object Definition

```
#OBJ
      object_id = id_9823_t
      discipline = drive_train
      group = transmission
      color = (0.5 0.6 0.7)
      .
      .
#END
```

Binary Object Definition

```
0002000200000009696 45F393832335F74—> #OBJ(2) object_id(2)
|  ||  ||      ||                |9 'id_9823_t'
      00000000000B64726976655F747261696E—> discipline(0)
      |  ||      ||                    |11 'drive_train'
      00010000000C7472616E736D697373696F6E—> group(1) 12
'transmission'
      |  ||      ||                      |
      000300033FFE00003FFE33333FFE6666—> color(3) literal(3)
      |  ||  ||      ||      ||      |0.5 0.6 0.7
```

# Geometry Specifications

An object can have any number of geometry specifications. The following is the format for the geometry specifications section:

geometry_attribute_number
number_of_geometries
geometries

In this format, geometry_attribute_number is a 2-byte integer equal to 5, and number_of_geometries is a 4-byte integer.

There are three types of geometries as follows:

- flat polygon
- shell polygon
- polyline

The shell format defines polygons as a network of interconnected polygons that share common points resulting in a unit typically called a polyhedron. Shells can be smooth shaded by EPD.Visualizer but flat polygons cannot.

## Flat Polygons

The following is the format for the flat polygon definition:

flat_polygon_type_number
number_of_points
point_list

In this format, flat_polygon_type_number is a 2-byte integer equal to 0, number_of_points is a 4-byte integer with a value indicating the number of points in the polygon, and point_list is a list of polygon points. A polygon point is defined

using three 4-byte single precision floating point values that represent an xyz location. Flat polygons must have a minimum of three points to define a plane.

The following example shows a textual flat polygon definition and a corresponding binary flat polygon definition in hexadecimal notation:

Textual Flat Polygon Definition

```
#FP
      10.0  0.0  0.0
      20.0  0.0  0.0
      20.0  5.0  0.0
      10.0  5.0  0.0
#END
```

Binary Flat Polygon Definition

```
00000000004400240000000000000000000—> #FP(0) 4 10.0 0.0 0.0
|  ||      ||      ||      ||      |
      40034000000000000000000—>20.0 0.0 0.0
      |      ||      ||      |
      40034000400140000000000—>20.0 5.0 0.0
      |      ||      ||      |
      40024000400140000000000—>10.0 5.0 0.0
      |      ||      ||      |
```

## Shell Polygons

The following is the format for the shell polygon definition:

shell_polyhedron_type_number
point_specification
face_descriptions

In this format, shell_polyhedron_type_number is a 2-byte integer equal to 1. The point_specifications define a list of points used to make up a shell, and the face_descriptions definition identifies which points to use when defining a face.

The following is the format for the point specification definition:

number_of_points
point_list

In this format, number_of_points is a 4-byte integer with a value indicating the number of points in the point list, and point_list is a list of three 4-byte single precision floating point values that represent an xyz location.

The following example shows a textual point specification definition and the corresponding binary point specification definition in hexadecimal notation:

Textual Point Specification Definition

```
#POINTS
      70.5  -20.9  0.5
      60.0  -10.0 -5.0
      90.0  -15.0 -5.5
      30.2  -10.1  0.0
      15.0   20.0  2.0
      10.7   30.2  2.4
      10.0    0.0  0.0
#END
```

Binary Point Specification Definition

```
0000000740051A00C034E6663FFE0000—> 770.5  -20.9  0.5
|      ||      ||      ||      |
      4004e000C0240000C0140000—>60.0  -10.0 -5.0
      |      ||      ||      |
      40056800C02E0000C0160000—>90.0  -15.0 -5.5
      |      ||      ||      |
      4003E333C024333300000000—>30.2  -10.1  0.0
      |      ||      ||      |
      4002E000400340004000000—>15.0   20.0  2.0
      |      ||      ||      |
      400256664003E33340003333—>10.7   30.2  2.4
      |      ||      ||      |
      400240000000000000000000—>10.0    0.0  0.0
      |      ||      ||      |
```

The following is the format for the face descriptions definition:

face_list_length
face_list

In this format, face_list_length is a 4-byte integer with a value indicating the number of numbers in a face list, and face_list is a list of 4-byte integers identifying which points to use when defining a face.

The following example shows a textual face descriptions definition and a corresponding binary face descriptions definition in hexadecimal notation:

Textual Face Description Definition

```
#FDS
      4  0  1  2  3
      4  2  3  4  6
      5  3  0  2  4  5
      -4  2  5  3  0
#END
```

Binary Face Description Definition

```
00000015000000040000000000000001000000200000003—> 214 0 1 2 3
|      ||      ||      ||      ||      ||      |
      00000004000000020000000300000004000000006—>4 2 3 4 6
      |      ||      ||      ||      ||      |
      00000005000000030000000000000002000000040000005—>5 3 0 2 4
5
      |      ||      ||      ||      ||      ||      |
      FFFFFFFC000000020000000500000003000000000—>-4 2 5 3 0
      |      ||      ||      ||      ||      |
```

## Polyline

The following is the format for the polyline definition:

polyline_type_number
number_of_points
point_list

In this format, polyline_type_number is a 2-byte integer equal to 4, number_of_points is a 4-byte integer with a value indicating the number of points in the polyline, and point_list is a list of polyline points. A polyline point is defined with three 4-byte single-precision floating-point values that represent an xyz location. Polylines must have a minimum of two points to define a line.

The following example shows a textual polyline definition and a corresponding binary polyline definition in hexadecimal notation:

Textual Flat Polyline Definition

```
#PL
      10.0  0.0  0.0
      20.0  0.0  0.0
      20.0  5.0  0.0
      10.0  5.0  0.0
#END
```

Binary Flat Polyline Definition

```
00040000000440024000000000000000000—> #FPL(4))4 10.0 0.0 0.0
|  ||      ||      ||      ||      |
      400340000000000000000000—>20.0 0.0 0.0
      |      ||      ||      |
      400340004001400000000000—>20.0 5.0 0.0
      |      ||      ||      |
      400240004001400000000000—>10.0 5.0 0.0
      |      ||      ||      |
```

## Textual and Binary Graphic File Example

The following example shows a complete textual graphic file and a corresponding binary graphic file in hexadecimal notation. The textual graphic file follows:

```
#GRAPHICS_FILE
      #CONTROL_OPTIONS
            unit = inches
      #END
            #EXTENTS
                        xmin = -10.0
                        ymin = -10.0
                        zmin = -10.0
                        xmax =  10.0
                        ymax =  10.0
                        zmax =  10.0
            #END
                  #COLORS
                              red   = (1.0  0.0  0.0)
                              green = (0.0  1.0  0.0)
                              blue  = (0.0  0.0  1.0)
                  #END
            #OBJ
                  object_id = -2124343233
                  discipline = structural
                  group = 100
                  color = blue
             #FP
                  10.0  0.0  0.0
                  20.0  0.0  0.0
                  20.0  5.0  0.0
                  10.0  5.0  0.0
             #END
             #SP
                  #POINTS
                        70.5  -20.9  0.5
                        60.0  -10.0 -5.0
                        90.0  -15.0 -5.5
                        30.2  -10.1  0.0
                        15.0   20.0  2.0
                        10.7   30.2  2.4
```

```
                        10.0    0.0  0.0
                  #END
                  #FDS
                        4  0  1  2  3
                        4  2  3  4  6
                        5  3  0  2  4  5
                        -4  2  5  3  0
                  #END
             #END
            #END
#END
```

# Binary

```
FF00030000000100060002 —> DECstation(FF) #CONTROL_OPTIONS(3) 1
unit(6)
|||  ||      ||  ||  |    inches(2)
0000C0240000C0240000C0240000—> #EXTENTS(0)-10.0 -10.0 -10.0
|  ||      ||      ||      |
      400240004002400040024000—>             10.0  10.0  10.0
      |      ||      ||      |
000100000003            —> #COLORS(1)     number of colors(3)
|  ||      |
      000000037265643FFF00000000000000000000    —> name length(3)
      |      ||    ||      ||      ||      |          'red' 1.0
0.0 0.0
      00000005677265656E000000003FFF000000000000—> name length(5)
      |      ||        ||      ||      ||      |      'green' 0.0
1.0 0.0
      00000004626C7565000000000000000003FFF0000  —> name length(4)
      |      ||      ||      ||      ||      |        'blue' 0.0
0.0 1.0

000200020000000B2D32312334333433323333—>  #OBJ(2) object_id(2)
|  ||  ||      ||                    |    11 '-2124343233'
      00000000000A7374727563747572616C—>  discipline(0) 10
'structural'
      |  ||      ||                  |
      000100000003313030            —>    group(1) 3 '100'
      |  ||      ||    |
      000300020000004626C75653      —>    color(3) symbolic(2) 4
'blue'
      |  ||  ||      ||       |
000500000002                        —>    polygon section(5)  2
|  ||      |
00000000000440024000000000000000000—>    #FP(0) 4 10.0 0.0 0.0
|  ||      ||      ||      ||      |
        400340000000000000000000    —>    20.0 0.0 0.0
        |      ||      ||      |
        400340004001400000000000    —>    20.0 5.0 0.0
        |      ||      ||      |
        400240004001400000000000    —>    10.0 5.0 0.0
```

```
        |      ||      ||      |
00010000000740051A00C034e6663FFE0000 —> #SP(1) 7  70.5  -20.9  0.5
|  ||      ||      ||      ||      |
        4004e000C0240000C0140000     —>           60.0  -10.0 -5.0
        |      ||      ||      |
        40056800C02e0000C0160000     —>           90.0  -15.0 -5.5
        |      ||      ||      |
        4003E333C024333300000000     —>           30.2  -10.1  0.0
        |      ||      ||      |
        4002E00040034000400000000    —>           15.0   20.0  2.0
        |      ||      ||      |
        400256664003E33340003333     —>           10.7   30.2  2.4
        |      ||      ||      |
        40024000000000000000000      —>           10.0    0.0  0.0
        |      ||      ||      |
000000150000000400000000000000001000000020000000       —> 214 0 1 2
3
|       ||      ||      ||      ||      ||      |
      00000004000000020000000300000004000000006        —>4 2 3 4 6
      |      ||      ||      ||      ||      |
      00000005000000030000000000000002000000040000000 5—>5 3 0 2 4
5
      |      ||      ||      ||      ||      ||      |
      FFFFFFFC000000020000000500000030000000 0        —>   -4 2 5
3 0
      |      ||      ||      ||      ||      |
```

Appendix D

# Textual and Binary Graphic File Conversion

This appendix describes the commands used to convert textual graphic files to binary graphic files, and binary graphic files to textual graphic files. These commands are the following:

- `pvs_atob` converts EPD.Visualizer textual graphic files (`.gaf`) to binary graphic files (`.gbf`)
- `pvs_btoa` converts EPD.Visualizer binary graphic files (`.gaf`) to textual graphic files (`.gbf`)

The options include converting the model from y-up to z-up vertical orientation, specifying the output file name, and silencing the conversion report on standard error.

Formats for path, Gpath, light, and material definitions are not supported for binary graphic files. A summary of any unconverted data is provided after the command is complete.

The following conventions are used for options in command syntax:

- No Brackets — Must be used once and cannot be repeated
- Square Brackets [ ] — Optional information that can only be used once
- Braces { } — Items listed in braces are exclusive (only one can be used)

Please note: Items within braces might or might not be required, according to bracket/no bracket conventions.

## Syntax

The following is the syntax format for `pvs_atob` and `pvs_btoa`:

```
pvs_atob [-s] [-y] [-o <file name>] [<source file name>]
pvs_btoa [-s] [-y] [-o <file name>] [<source file name>]
```

The following list explains the options from the preceding format:

- `-s` — Silence the conversion report on standard error
- `-y` — Convert the model from y-up to z-up vertical orientation
- `-o` — Specify the output file name

The following should be taken into consideration with this syntax:

- If you do not use the `-o` option, the standard output is used.
- If no source file is specified, the standard input is used.
- If you enter an invalid option, the syntax format for these commands displays.
- Formats for path, Gpath, light, and material definitions are not supported for binary graphic files. When the `pvs_atob` command encounters these formats, it checks for the syntax, but does not generate corresponding binary graphic file information. A summary of this unconverted data is provided after the command is complete.

## Examples

The following example shows the command to convert a textual graphic file to a binary graphic file and specifies the name of the output file:

```
pvs_atob -o project1.gbf project1.gaf
```

The following example shows the command to convert a textual graphic file and direct it to a binary output file:

```
pvs_atob project1.gaf > project1.gbf
```

The following example shows the command to pipe the output of one process as the input to another process:

```
cadds2pvs -s caddspart/_td | pvs_atob -o caddspart.gbf
```

Appendix E

# DIMENSION III Interface

The DIMENSION III interface provides you with the capability to display full-scale DIMENSION III models on EPD.Visualizer. You can process a DIMENSION III viewfile with this interface, transfer the files, and then review your project using EPD.Visualizer. The following topics are discussed in this appendix:

- Introduction
- Generating Graphic Files
- ASCII Graphic File Contents
- Error Messages

# Introduction

The DIMENSION III interface, PVGENPOLY, is distributed with the DIMENSION III product. PVGENPOLY generates ASCII or binary graphic files from DIMENSION III viewfiles. The ASCII and binary files contain the same information, but have different advantages. The ASCII files are compatible with all systems, but the binary file is processed significantly faster than an ASCII file. However, binary files are not compatible with all systems.

DIMENSION III viewfiles created on VAXstations have compatible binary files with DECstations running ULTRIX. DIMENSION III viewfiles created on HP/Apollos have compatible binary files with Sun SPARCstations, HPs, and Apollos. When you transfer binary files to a different workstation using FTP, you must set the transfer mode to binary using the binary command.

# Generating Graphic Files

You can generate graphic files for one or more DIMENSION III viewfiles. To process multiple viewfiles, enter the commands in an AEGIS script file or a VAX command file and then execute the file.

The following sections provide the steps for processing single or multiple viewfiles.

## Processing Single Viewfiles

To process single viewfiles, at your system prompt enter:

```
pvgenpoly
```

You are prompted for the viewfile name with the following:

```
Enter viewfile name:
```

You can enter either an entire path name or the viewfile name, or you can press `Return` to exit the interface.

After you enter the viewfile name (with or without the `.vf` extension), press `Return`. If you do not use the `.vf` extension, it is automatically appended to the end of the viewfile name. You are prompted with the following:

Menu 1

```
Specify file type to generate
      1 = ASCII file
      2 = binary file
      3 = both file types
Enter 1,2 or 3:
```

After you select an option, press `Return` and you are prompted with the following:

Menu 2

```
Specify unit type
      1     =     mils
      2     =     inches
      3     =     feet
      4     =     yards
      5     =     miles
      6     =     millimeters
```

```
      7     =     centimeters
      8     =     decimeters
      9     =     meters
      10    =     kilometers
Enter number for unit type:
```

After you select an option, press `Return`. The following message displays:

```
Generating graphic file(s)...
```

After the graphic files are generated, the following message displays:

```
Graphic files are successfully generated!
```

If you selected option 1 on Menu 1, an ASCII graphic file is generated. The new graphic file name is the same as the viewfile name, appended with a `.gaf` extension.

If you selected option 2 on Menu 1, a binary graphic file is generated. The new graphic file name is the same as the viewfile name, appended with a `.gbf` extension.

If you selected option 3 on Menu 1, both an ASCII and a binary graphic file are generated. The new files names are the same as the viewfile names, appended with a `.gaf` extension or a `.gbf` extension, respectively.

After the viewfiles are processed, you are prompted with the following:

Menu 3

```
Do you want to generate more graphic files? (y/n):
```

If you enter `y`, Menu 1 displays again. If you enter `n`, you return to the system prompt.

## Processing Multiple Viewfiles

To process multiple viewfiles, you can create a command file or shell script using the in-line command input. An unlimited number of viewfiles can be processed. On the line after the viewfile name or path name, enter either 1, 2, or 3 to respond to Menu 1. The next line must contain the unit type to respond to Menu 2. The last line must contain either `y` or `n` to respond to Menu 3.

On VMS, you can create a command file using the DECK and EOD in-line commands. The first two lines and the last line of the command file must contain those shown in the following example:

```
$pvgenpoly
$DECK
<viewfile name 1>
1
3
y
<viewfile name 2>
2
7
y
<viewfile name 3>
3
4
n
$EOD
```

To execute the command file, at the system prompt enter:

```
@<command file name>
```

On AEGIS, the first and last line of the script file must contain those shown in the following example:

```
pvgenpoly<<LABEL
<viewfile name 1>
1
9
y
<viewfile name 2>
2
1
y
<viewfile name 3>
3
5
n
LABEL
```

To execute the script file, at the system prompt enter the script file name.

# ASCII Graphic File Contents

The ASCII graphic files generated from DIMENSION III viewfiles contain the following information:

- `#CONTROL_OPTIONS` block — Contains the unit type you selected when you generated the graphic file (such as, inches, millimeters).
- `#COLOR` block — Contains definitions for eight colors. The color numbers, names, and RGB values for DIMENSION III items are mapped to the colors shown in Table E-1.

**Table E-1 DIMENSION III Color Map**

| Color Number | Color Name | RGB Value |
|---|---|---|
| 0 | black | 0.00.00.0 |
| 1 | red | 1.00.00.0 |
| 2 | green | 0.01.00.0 |
| 3 | yellow | 1.01.00.0 |
| 4 | blue | 0.00.01.0 |
| 5 | magenta | 1.00.01.0 |
| 6 | cyan | 0.01.01.0 |
| 7 | gray | 0.50.50.5 |

- `#OBJ` block — Each object block corresponds to a DIMENSION III item. Each `#OBJ` block contains the following attributes:
- Group Name — The item's class number.
- `Object_ID` — A combination of the ID pointer and the model ID number (extended ID pointer)
    - Color Name — A name corresponding to the item's color number specified in Table E-1. You can edit the color definitions in an ASCII graphic file to contain different RGB values.
    - Polygon Definitions — The polygon definitions define the object's geometry.

All DIMENSION III triangulated surface items are generated as shell polygons. These items can be smooth shaded by selecting the smooth shading rendering option in EPD.Visualizer. All other items are represented by flat polygons. If you want smooth shading for items with flat polygons, EPD.Visualizer provides a utility program named smooth that converts flat polygons to shell polygons. Smooth resides in `/usr/apl/CVpvs/bin`.

# Error Messages

- `Error in opening or reading viewfile!`

  You did not enter a valid viewfile name.

- `Invalid option! Try again.`

  You did not enter a valid option number.

- `Error in opening ASCII file: <`file name`>!`

  A problem exists at the operating system level (for example, there might be a protection problem). Notify your System Administrator for assistance.

- `Error in opening binary file: <`file name`>!`

  A problem exists at the operating system level (for example, there might be a protection problem). Notify your System Administrator for assistance.

- `No item found in viewfile!`

  The viewfile does not contain any items.

- `Error in writing polygon data!`

  A problem exists at the operating system level (for example, there might be a protection problem). Notify your System Administrator for assistance.

- `Error in removing ASCII polygon file!`

  A problem exists at the operating system level (for example, there might be a protection problem). Notify your System Administrator for assistance.

- `Error in removing binary polygon file!`

  A problem exists at the operating system level (for example, there might be a protection problem). Notify your System Administrator for assistance.

Appendix F

# CADDS Interface

With the CADDS interface you can display CADDS 5 and CADDS4x parts or assemblies in EPD.Visualizer. With this interface, you can process CADDS parts or assemblies, transfer the files, then review them with EPD.Visualizer. The CADDS interface also provides a command for rotating or translating information. You can use the CADDS interface at the UNIX level on the HP, Silicon Graphics and Sun systems. The following topics are discussed in this appendix:

- Introduction
- Using the CADDS Interface
- Reorienting Objects in Graphic ASCII Files

Please note: This interface is distributed only with CADDS systems.

# Introduction

The CADDS interface, `cadds2pvs`, converts tessellation data from CADDS 5 and CADDS4x parts or assemblies into graphic ASCII files or graphic binary files. It accepts two styles of input. You can provide a CADDS part name for `cadds2pvs`, or you can use the command as a filter by piping CADDS tessellation data into it. Output can go to standard error, or you can redirect it to a file. Messages go to standard error, which is usually your screen.

The SHADE PICTURE command creates the CADDS tessellation data, the `_td` file, in the directory of the CADDS part. In the same directory, the `_pd` file stores part data information and the `_db` files stores assembly design data.

If required, you can use the `transpvs` command to rotate or translate information in the graphic ASCII files. Refer to “Reorienting Objects in Graphic ASCII Files” on page F-11 .

# Using the CADDS Interface

The following is a description of how to use the CADDS interface, `cadds2pvs`, followed by some examples.

The following conventions are used for modifiers in command syntax:

- No Brackets — Must be used once and cannot be repeated
- Square Brackets [ ] — Optional information that can only be used once
- Braces { } — Items listed in braces are exclusive (only one can be used)

Please note: Items within braces might or might not be required, according to bracket/no bracket conventions.

The first step to using `cadds2pvs` interface is to convert CADDS parts or assemblies to graphic ASCII files or graphic binary files with the following command syntax.

```
cadds2pvs [-a adrawing] [-b] [-c] [-C comp_id]
      [-d drawing_name] [-e] [-f] [-o file_name]
      [-s] [-S] [-t tolerance] [-u unit_type]
      [-v tolerance] [-w]
      [CADDSpart ...]
```

The options in the preceding example are explained in the following list:

- `-a` — Obtain layer colors from selected assembly `adrawing`. There should be a `_pd` file in a corresponding directory.
- `-b` — Output graphic binary files (`.gbf`) instead of ASCII.
- `-c` — Convert assembly components into separate files in the form of `dir/c_i.x`, where:
  - `dir` = the directory portion from the `-o` option (or `pwd`)
  - `c_i` = the name of the component instance
  - `.x` = `.gaf` or `.gbf`
  - Any `&` characters in `c_i` are converted to `~`. The `-o` file (or standard output) receives any non-component output.
- `-C` — Process only the indicated assembly component. Use the `-c` option with this option.
- `-d` — Obtain ECHO LAYER information from selected drawing, rather than arbitrary drawing. Use the `-e` option with this option.
- `-e` — Restrict output to echoed layers only.

- `-f` — Convert `#SP` blocks to `#FP` blocks.
- `-o` — Output file name, instead of standard output.
- `-s` — Silence the progress report on standard error.
- `-S` — Convert `#FP` to `#SP` blocks. Normally, `cadds2pvs` tries to create `#SP` blocks where needed.
- `-t` — Eliminate every facet edge in a shell polygon whose sum of xyz components is less than the specified tolerance. The default is 1.0. It must be expressed in millimeters. Use 0 for no elimination. See the `-v` option.
- `-u` — Identify the units of tessellation data when a `_pd` file is unavailable. The following are the valid units:

  mils

  inches

  feet (default)

  yards

  miles

  millimeters

  centimeters

  meters

  decimeters

  kilometers
- `-v` — Tolerance for flat or shell polygon vertices, which should be identical. The default is 0.01. See the `-t` option.
- `-w` — Formats the output with extra white space.
- `CADDSpart` — Path name of your CADDS part directory. Alternatively, it might be the name of an assembly created by Assembly Design. You can supply more than one.

# Examples

The following is an example of the basic use of the `cadds2pvs` interface under UNIX.

1. Change to the directory containing the CADDS part by entering:

   `cd` directory_path_name

   For example:

   ```
   cd /users/joe/parts/demo/equipment
   ```

2. List the directory contents by entering

   ```
   ls
   ```

   The following is a sample of the directory contents:

   ```
   _pd    _td    drawing1
   ```

3. Enter the interface command and options as the following example shows:

   ```
   cadds2pvs -o ../equipment.gaf .
   ```

   `cadds2pvs` reads the `_td` and `_pd` files in the directory and generates a file named `equipment.gaf` in the demo directory.

If you want to have the graphic ASCII file reside in the CADDS part directory, follow the steps in this example:

1. Change to the directory containing the CADDS part by entering:

   ```
   cd /users/joe/parts/demo/equipment
   ```

2. Enter the interface command and options as follows:

   ```
   cadds2pvs -o gaf .
   ```

For standard output, follow the steps in this example:

1. Change to the directory containing the CADDS part by entering:

   ```
   cd /users/joe/parts/demo/equipment
   ```

2. Enter the interface command and options as the following example shows:

   ```
   cadds2pvs . > gaf
   ```

To avoid confusion when you move the `.gaf` file to another directory, you can create a symbolic link as the following example shows:

```
ln -s /users/joe/parts/pvsdemo/equipment/gaf \
                              /usr/pvsdemo/equipment.gaf
```

## Restrictions

The following restrictions apply when using the `cadds2pvs` interface:

- If standard input is used, you can supply the contents of only one `_td` file.
- When colors for layers are unavailable (because the `_pd` is to be ignored or SELECT LDISCRIMINATION was never performed), then colors are assigned by the following formula:

  ```
  color# = (layer# modulo 64) + 1
  ```

- If some surfaces are not treated as backfaces, turn off the EPD.Visualizer option View > Rendering... Backfacing Polygons. If your graphic ASCII file was produced with the help of a `_pd` file, then this should be considered a bug and you should report it.
- If the model was designed in Parametric Mode, then use the CHANGE ENTITY command in CADDS to convert entities to explicit. This action permits `cadds2pvs` to read surface normal flags to improve the results of surface culling in EPD.Visualizer.

  Remember to follow the PACK DBASE and SORT DBASE commands with the SHADE PICTURE command to update the `_td` file.
- Currently, the Digital version of `cadds2pvs` cannot read `_pd` files. Therefore, use the `-u` option. However, with this option, colors and the Backfacing Polygons option might suffer.
- Assembly Design might only work on some platforms, such as Sun and HP. Also, if the `ODB_SERVER` is the wrong revision, it might not work well with `cadds2pvs`. Check the version of CADDS 5i or CADDS4X with which the `ODB_SERVER` is normally used. The `cadds2pvs` version 2.0 relates to an `ODB_SERVER` for CADDS 5 Rev 4.1 or later, while cadds2pvs 1.0 applies to earlier versions of `ODB_SERVER`.
- To reduce the size of a graphic ASCII file, consider using lower values for the SMOOTH option to the SHADE PICTURE command. You can also use The UNIX command `compress` for long term storage of graphic ASCII files.

## Messages and Errors

If an illegal option is supplied, help is provided on standard error. For example, enter the following command to display a description of how to use this command while in `csh`:

```
cadds2pvs -help | & more
```

The following informational messages are produced, which you can suppress using the `-s` option:

- `Units are` **xxxx**

  As a `_pd` file is successfully read, its model units are displayed.

- `No layer/color discrimination (assigning)`

  Refer to “Restrictions” on page F-6 . Colors are assigned automatically.

- **xxxxxxx** `bytes read`

  The number of bytes successfully read from the `_td` file are dynamically displayed. When the command is done, the result matches the total byte size of the `_td` file.

- `Reading` **xxxx**`/_pd, Reading` **xxxx**`/_td, Reading stdin`

  These messages indicate that these files are being successfully read.

- `Warning,` **xxxx**`/_pd unavailable`

  This message indicates that the `_td` will be used without the benefits of a `_pd` file.

- Cannot open **xxxx**

  When this message pertains to a `_td` file, then the path name contains a misspelled directory level, or you do not have read privileges for that file. The command proceeds to the next CADDS part, if any.

- `Warning:` **xxxx**`/_td is inconsistent with its _pd`

  The `_td` refers to entities that no longer exist in the `_pd` file. This usually means that the CADDS part was modified, packed, or sorted without performing SHADE PICTURE before filing.

The following error messages might display and return you to the system prompt:

- `Cannot open` **xxxx**

  When this message pertains to the graphic ASCII file, then the path name contains a misspelled directory level or you do not have write privileges for that file.

- Unit **xxxx** was invalid

  Use the units names described in the `SYNTAX SUMMARY`.

- `Option` **xxxx** `was not recognized`

  An option with a leading `-` (hyphen) was invalid.

- `Error at` **xxxx**`:`**xxxx**`, Aborted at byte position` **xxxx** `(0xhhhh) in tessellation data`

`cadds2pvs` cannot interpret a section of the `_td` file. This should not occur unless the `_td` file is corrupt. If you believe your `_td` is correct, supply the message and the version of `cadds2pvs` in a bug report.

- `Premature end of file encountered!`

  The `_td` file appears to be truncated.

## Converting Tessellation Data Only

Although the `cadds2pvs` interface does not have mandatory options, it is recommended that you supply one or more CADDS part names on the command line. In that case, the `_td` file and `_pd` file are both used to generate the output file.

When no CADDS part name is supplied, then the contents of a `_td` file is required as standard input and the information from a `_pd` file is lost. Similarly, if the `_pd` is missing in the CADDS part directory but the `_td` exists, the parts are processed and the following message displays:

```
Warning: equipment/_pd unavailable
```

The `_pd` file contains the following information that cannot be obtained from the `_td` file:

- The model units of the part (meters, inches, etc.)
- The colors of CADDS layers (from SELECT LDISCRIMINATION)
- Surface normal flags (to improve backface culling)
- Relationships between solids and surface entities

Without the `_pd` file, you might encounter some minor difficulties. The following are three examples that avoid the `_pd` and perform equivalent tasks:

```
cat equipment/_td | cadds2pvs -o equipment.gaf -u feet

cadds2pvs -u feet -o equipment.gaf < equipment/_td

cadds2pvs -u feet < equipment/_td > equipment.gaf
```

Please note: The `-u` option is used in these examples. You must specify the units manually otherwise, the `_pd` file will be ignored.

# Multiple Input and Output

Now that you have learned how to use `cadds2pvs`, you can use system level commands in conjunction with it. To generate one graphic ASCII file for several parts, follow the steps in this example:

1. Change to the directory containing the CADDS part by entering:

   ```
   cd /users/joe/parts/demo
   ```

2. List the directory contents by entering:

   ```
   ls
   ```

   The following is a sample of the directory contents:

   ```
   arch equipment hvac piping site structure
   ```

3. Enter the interface command and options as the following example shows:

   ```
   cadds2pvs -o model.gaf *
   ```

To generate individual graphic ASCII files for selected CADDS parts with `csh`, enter the following command:

```
foreach part (arch equipment hvac piping site structure)
      echo Processing $part ======================
      cadds2pvs -o $part.gaf $part
end
```

To generate individual graphic ASCII files for all CADDS parts in all subdirectories with `sh`, enter the following command:

```
sh% find /users/joe/parts -name _pd -print |\
      sed sX/_pdXX |\
      while read part; do
         echo Processing $part================
         cadds2pvs -o $part.gaf $part
      done
```

To use a remote shell to run `cadds2pvs`, creating the graphic ASCII file locally on a system, enter the following command:

```
remsh foo -n cadds2pvs /usr/tmp/house > house.gaf
```

To use a remote shell to feed a `_td` file to your local system for conversion, enter the following command:

```
remsh bar -n cat /usr/tmp/car/_td | cadds2pvs > car.gaf
```

Please note: The `remsh` command is the UNIX `rsh` command on Sun and Digital systems.

## Assembly Design Trees

You can process assemblies that were created by Assembly Design commands. In place of the CADDSpart option, supply the name of your assembly.

Set the CVPATH UNIX environment variable, just as the .caddsrc file would set it. It is used for locating the components of an assembly tree. (If you forget to set CVPATH, then cadds2pvs supplies a default value.) The following is the syntax for setting the CVPATH:

```
setenv CVPATH directory : directory : directory
```

In addition, you can override the following defaults:

```
setenv DB_DAEMON_HOST `/bin/hostname`
setenv ODB_am_wh /usr/apl/cadds/bin/ODB_SERVER
setenv DB_DAEMON 20000001
```

The DB_DAEMON_HOST system must be running the ODB_DAEMON process before using the cadds2pvs command on assemblies. Remember to use the SHADE PICTURE command to generate _td files for the components in your assembly.

# Reorienting Objects in Graphic ASCII Files

In some instances, your CADDS model might involve viewing other parts with arbitrary orientations and locations without the use of Assembly Design. Because references to the viewed parts are not stored with your model, `cadds2pvs` cannot attempt to locate them. To adjust a graphic ASCII file for a specified rotation and location, use the following `transpvs` command (rotates or translates objects in a graphic ASCII file).

`transpvs [-u unit type] [-w] AX AY AZ X Y Z <` old .gaf `>` new .gaf

## Options

Options for the `transpvs` commands are as follows:

- `-u` — The units for the xyz offset. The default units are the units specified in the `old.gaf` file. The following are the valid units:

  mils

  inches

  feet

  yards

  miles

  millimeters

  centimeters

  meters

  decimeters

  kilometers

  If you do not enter the unit type, the xyz offsets maintains the same units as the `old.gaf` file.
- `-w` — Formats coordinate data.
- `AX AY AZ` — Three angles (in degrees) to rotate the coordinate data about the x axis, then the y axis, and then the z axis.
- `X Y Z` — Three numbers to translate the coordinate data along the `x`, y, and z directions; see the `-u` option.
- `old.gaf` — Old graphic ASCII file name.
- `new.gaf` — New graphic ASCII file name.

Please note: The `old.gaf` and the `new.gaf` files must have different names.

## Examples

The following example shows that the coordinates in the file `equipment.gaf` will first rotate about the x axis by 90 degrees and then about the y axis by 180 degrees. It is then translated by 100 meters along the x axis and 300 meters along the z axis. The resulting output is directed to a file called `equip.upper.gaf`.

```
transpvs -u meters 90 180 0 100 0 300 < equipment.gaf >
                                        equip.upper.gaf
```

## cadds2pvsdb

A UNIX Bourne shell script that reads graphic ASCII files produced by `cadds2pvs` and generates the contents of a EPD.Visualizer database. You can alter the script to suit your needs.

A suggested format of a `pvs` database definition file (`.def` file) is mentioned by comments within the script.

The syntax for `cadds2pvsdb` is as follows:

```
cadds2pvsdb file.gaf ...
```

the option for `cadds2pvsdb` is the following:

`file.gaf` — One or more graphic ASCII files produced by `cadds2pvs`.

Please note: Because the value of `Object_ID` in the graphics ASCII files created by `cadds2pvs` is now numeric, special decoding is required. This is accomplished by understanding that a model sequence number resides in the high order 16 bits (starting from -32768) and a master index pointer (MIPTR) in the low 16 bits. In the C language, this is described with the formula:

```
object_id = ((-32768+model_seq_number) <<16) + miptr
```

Appendix G

# EPD.Visualizer Movie Player

This appendix provides information about the EPD.Visualizer movie player, `pvsmovie` and includes details on the following:

- Getting Started
- Loading a Movie
- Quitting pvsmovie
- Playing a Movie

This application enables you to view image sequences captured during the item and camera traversals at a much higher speed than real time rendering.

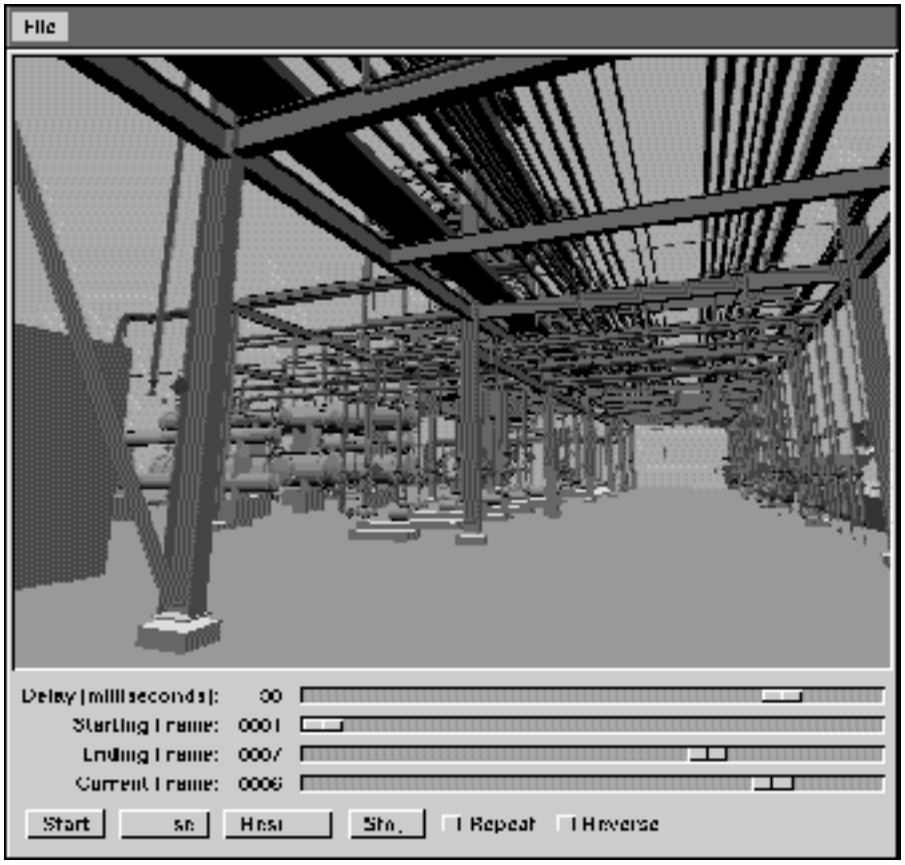
To replay a movie sequence on the Silicon Graphics, you can select RGB format in the EPD.Visualizer Movie Recorder dialog and use the movie player provided with all IRIX systems. On the Sun, you can select RAS format and use the CV Animator provided with CADDS systems.

To run the EPD.Visualizer Movie Player, add the following directory to the platform-specific shared library path environment variable:

```
${VIS_HOME}/lib
```

For example, on platforms such as Sun, Compaq, or SGI, you would add this directory path to the `LD_LIBRARY_PATH` environment variable. On the HP platform, you would add this path to the `SHLIB_PATH` environment variable. For IBM, you would add it to the `LIBPATH` variable.

When executing `pvsmovie`, the EPD.Visualizer Movie Player window displays, as shown in the following figure:



# Getting Started

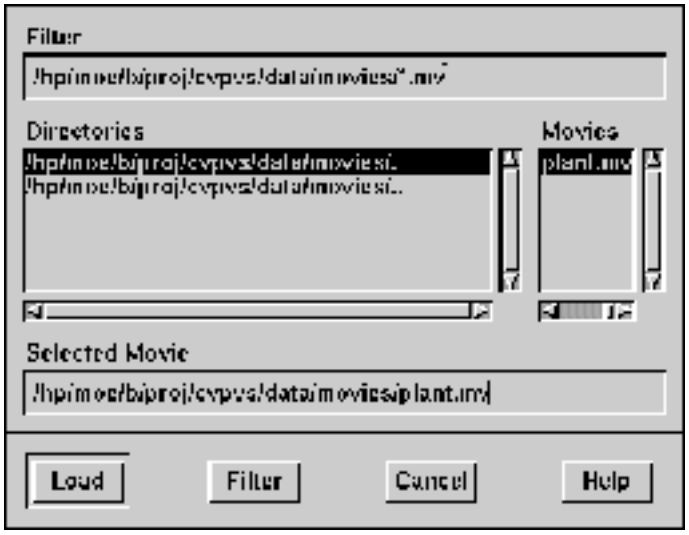
By selecting File in the menu bar, you can access the following menu items:

- Load — Load a movie file
- Quit — Quit EPD.Visualizer

# Loading a Movie

To load a movie file, use the following steps:

1. Select File > Load. The following figure shows the EPD.Visualizer Movie Selection dialog box:




2. Double click a file name in the Movies list, or enter a file name in the Selected Movie box and click Load. The default extension for movie files is `.mv`. These files are generated by the EPD.Visualizer Movie Recorder.

   Refer to “File Selection Dialog Box” on page 1-17 for more information on using the filter option.

3. Click OK.

   `pvsmovie` loads all the movie frames specified in the movie file into memory. This might take several minutes.

   The amount of memory used is approximately the sum of the sizes of all the movie files. If your system does not have adequate memory, it will take significantly longer to load and play the movie sequence.

   EPD.Visualizer Movie Recorder records movie files using 256 colors. If your system’s colormap does not have enough color cells to handle this, the system displays warning messages and the movie player adjusts the color usage accordingly.

   On any platform, it is possible during `pvsmovie` loading to receive the following error message printed in your startup window:

```
Error! Unable to allocate all colors!
```

The movie loads and plays successfully, however all colors do not display as originally created. When this occurs, exit the movie player, quit some of your color-intensive applications, and restart the movie player. This action should free the color table for your hardware.

On UltraSPARC, it might be your desktop environment (CDE) that is the most color intensive. In this case, switch to the OpenWindows environment.

# Quitting pvsmovie

To quit `pvsmovie`, select File > Quit.

# Playing a Movie

To play a movie sequence, use the following steps:

1. Load in a movie file. Refer to “Loading a Movie” on page G-4 for instructions on loading a movie.
2. Set the following options as necessary:
   - Delay

     The delay between each frame in milliseconds.
   - Starting Frame

     The number of the first frame to display.

   Ending Frame

     The number of the last frame to display.
   - Current Frame

     The number of the current frame displayed.
   - Repeat

     Repeat the sequence when finished.
   - Reverse

     When repeating a sequence, run the movie in reverse from the last frame to the first frame.
3. Click Start to play the movie sequence.
4. Click the following buttons as necessary:
   - Pause

     When Repeat is set, you can click Pause to temporarily pause at the current frame.
   - Resume

     Resume the movie playing from pause.
   - Stop

     Terminate playing the current movie sequence.

Appendix H

# Environment Variables

This appendix explains how you can use environment variables to customize the EPD.Visualizer. The following topics are discussed:

- General Behavior
- Graphics Files Loading
- EPD.Visualizer-EPD.Connect Communication
- Path Editor, Path Traversal, and Translation and Rotation
- PostScript Output
- Section Plane
- Zoning
- Selecting, Scrolling, and Running a Report

# General Behavior

The following examples show you how to set variables in a shell window.

Either of the following lines sets a non-zero value:

```
setenv ENVIRONMENT_VARIABLE_NAME
setenv ENVIRONMENT_VARIABLE_NAME 1
```

Either the following line or not writing any command to give the variable a value has the effect of setting a zero value:

```
setenv ENVIRONMENT_VARIABLE_NAME 0
```

The following environment variables configure general behavior.

## CVDPA_CONFIG_DIR

CVDPA_CONFIG_DIR specifies the directory used to save and retrieve the position of the dialog boxes memorized for each user. If the variable is not defined the positions of the dialog boxes are not stored. If the variable is defined the dialog box locations for each user are stored in a file whose name is formed by adding the suffix `.cfg` to the user name. For example, positions for the user `tester` are stored in the file `tester.cfg` in the directory specified by CVDPA_CONFIG_DIR.

If the variable is defined the dialog boxes display at the beginning of each session in the locations selected by each user, with the following behavior configured by permissions:

- If the `.cfg` file has read and write permissions open, the file stores new and altered dialog box positions.
- If the `.cfg` file has only read permission open, the file cannot store new or altered dialog box positions, and presents the same positions.

## HOOPS_PICTURE

HOOPS_PICTURE defines the Hoops driver used for display (for example, x11, xgl,...). See help on release in the Help menu for more information.

## PVS_ORTHO_MODE

PVS_ORTHO_MODE controls the default perspective mode. The mode is orthographic if PVS_ORTHO_MODE is defined and its value is not equal to `0`. The mode is perspective if PVS_ORTHO_MODE is not defined or its value is equal to `0`. The perspective mode specified by PVS_ORTHO_MODE is automatically set when EPD.Visualizer starts and at the beginning of every new session. See "Changing Views" on page 3-43.

## PVS_SECOND_VIEW_SIZE

PVS_SECOND_VIEW_SIZE defines the size of secondary views in relation to the main view, for example:

```
setenv PVS_SECOND_VIEW_SIZE 0.3
```

## PVS_STEREO

PVS_STEREO, If defined and not set to `0`, allows HOOPS to use stereo mode. See help on release in the Help menu for more information.

## PVS_STEREO_ANGLE

PVS_STEREO_ANGLE sets the stereo factor for stereo mode; its default value is `0.03`.

## PVS_TEMP_COLOR

PVS_TEMP_COLOR specifies the color used for the selection, markers, and temporary graphics, if you select Options > Temp Color > User from the EPD.Visualizer window.

The default value is black or white, according to the background color of the 3D scene. Black corresponds to a dark background color.

# PV_SYSTEM

PV_SYSTEM specifies the location of the user interface directory (uif). In an environment that does not include EPD.Connect, the usual location is `/usr/apl/CVpvs/common`.

In EPD.Connect the script that starts EPD.Visualizer automatically sets the value of PV_SYSTEM. The value depends on where EPD.Connect is installed, which is normally $EPD_HOME`/common`.

# Graphics Files Loading

The following environment variables configure graphics file loading.

## CVPATH

CVPATH specifies the search path used to find graphic files if the full name is not provided. CVPATH is usually defined in the `~/.caddsrc` file.

## PVS_CONCAVE_PERFORM

PVS_CONCAVE_PERFORM can reconfigure concave polygons to improve performance. If PVS_CONCAVE_PERFORM is set to `1` and the Check for Concave Polygons option is enabled in the Graphic File Selection dialog box, any concave polygon is converted into a set of triangles. See “Adding Textual or Binary Graphics Files” on page 3-5.

## PVS_CADDS_COLORS

PVS_CADDS_COLORS specifies the name of a text file associating colors and layers to which EPD.Visualizer refers when reading a tesselated data file. The file has the following format:

```
LAYER    COLOR
```

The elements of the format have the following characteristics:

- LAYER is 0-254 as a layer list without spaces, for example:

  ```
  1-20,50-73,90,100
  ```

- COLOR is number of a cadds color: 1-65.

If color file is not specified or does not exist, the selection is the default color 1 (white). An example file is provided in `/usr/apl/nav/data/explorer`.

# EPD.Visualizer-EPD.Connect Communication

The following environment variables configure EPD.Visualizer-EPD.Connect communication.

## CVDPA_USE_MIRFLAG

CVDPA_USE_MIRFLAG controls whether EPD.Visualizer is able to load components with a symmetry flag in addition to translation and rotation specifications. If the variable is set to any value, the EPD.Visualizer is able to load components in response to EPD.Connect requests with the X, Y, Z, AX, AY, AZ, and YMIRFLG parameters defined in the communication file. If the CVDPA_USE_MIRFLAG variable is not set, only X, Y, Z, AX, AY, AZ are used in loading the files.

YMIRFLAG is a field in the `LOAD` command, which has a syntax like the following:

```
LOAD graphic_file component X Y Z AX AY AZ YMIRFLAG
```

YMIRFLAG controls whether EPD.Visualizer loads components with a Y symmetry. If the field is set to `1`, the components are loaded with a Y symmetry (XZ plane symmetry). No symmetry is implied if the field is set to `0`.

## OLD_GLOBAL_SPELLING

OLD_GLOBAL_SPELLING assists in backward compatibility for command files generated by EPD.Visualizer for updating the product structure. If the variable is set to any value, the keyword `GLOBAL-ORIENT` is used instead of `GLOBAL_ORIENT` when EPD.Connect asks EPD.Visualizer to update the position of the components moved by the user. The keyword `GLOBAL-ORIENT` was used in Explorer 1.1.

## PVS_EXTERNAL

PVS_EXTERNAL specifies the name of file for communication client-to-pvs. Its default value is `/usr/tmp/client_pvs_comm_file`.

## PVS_CLIENT_FILE

PVS_CLIENT_FILE specifies the name of file for communication pvs-to-client. Its default value is `/usr/tmp/pvs_client_comm_file`.

# PVS_COLOR_FILE

PVS_COLOR_FILE specifies the name of a color table that provides the definitions of 16 colors that can be applied to blobs in the product structure tree. A table is provided in `/usr/apl/nav/data/explorer`.

EPD.Connect can change the color of an object in EPD.Visualizer by sending the following command through the communication process:

`Color` color_number name_of_component

The color_number is a number in the range 0-15 indicating the color in the EPD.Connect application. The file specified by PVS_COLOR_FILE file establishes the RGB values of the colors to be used in EPD.Visualizer. For EPD.Connect, a default table is supplied in `$EPD_HOME/data/exlorer/pvs_color_file`.

# Path Editor, Path Traversal, and Translation and Rotation

The following environment variables configure Path Editor, Path traversal, and translation and rotation behavior.

## PVS_LOCAL_CPL

PVS_LOCAL_CPL controls the default setting of the Local CPL toggle button in the Translate/Rotate Selection List dialog box. Setting the variable to any value specifies that the default setting of the toggle is active Local CPL. The setting affects sectioning from multiple Gpath references of the current selection.

## PVS_NORMAL_AXIS

PVS_NORMAL_AXIS specifies which axis of Gpath references is considered as normal to the material. Legal values are `x`, `y`, `z`, `X`, `Y`, or `Z`. The default value is `Z`.

## PVS_PATHS_DIR

PVS_PATHS_DIR specifies the default directory used to save the paths created from the Path Editor dialog box.

## PVS_UNNAMED_REF

PVS_UNNAMED_REF specifies the default string name used in creating names for new references. If the variable is not defined, the default string is `rf`.

# PostScript Output

The following environment variables configure PostScript output behavior.

## PVS_PRINT_COMMAND

PVS_PRINT_COMMAND specifies the command used to print the output file for Section Output and Graphics Output. Its default value is `lpr`. The command does not contain any parameter other than the file name.

## PVS_PRINT_FACTOR

PVS_PRINT_FACTOR specifies the scale factor applied to the PostScript output of the Section window to respect the user-defined scale. PVS_PRINT_FACTOR does not affect the output from the graphics window. Its value should be undefined or equal to `1.0` in most cases.

# Section Plane

The following environment variables configure section plane behavior.

## PVS_SECTION_AXIS

PVS_SECTION_AXIS specifies which axis of a reference is used to orientate the normal of the sectioning plane in motions to a reference. Legal values are `x`, `y`, `z`, `X`, `Y`, and `Z`. The default value is `X`.

## CVDPA_SECTIONS_DIR

CVDPA_SECTIONS_DIR specifies the default directory used when saving sections from the Sectioning - Save dialog box.

# Zoning

The following environment variables configure zoning behavior.

## CVDPA_ZONES_DIR

CVDPA_ZONES_DIR defines the directory used to store graphics files generated when creating a zone.

## CVDPA_SPACING

CVDPA_SPACING defines the distance between internal faces in zone definition.

# Selecting, Scrolling, and Running a Report

The following environment variables allow you to run a custom report on Select and Scroll:

## PVS_SELECT_SCROLL_REP

Set the variable `PVS_SELECT_SCROLL_REP` in the `explorer.ini` file to Y to enable you to run a custom report on Select and Scroll. The default value is N.

## CA_ENV_REPORT

The variable `CA_ENV_REPORT` defines the custom report that is to be run. This can be set in the following two ways:

- `CA_ENV_REPORT='LOOKUP:`<number>'

  In this case, the report is fetched from the `lookup.rep` file. The number is the number of the report to be run.

- `CA_ENV_REPORT='ACTION:[Report type]\t[Tree data wanted]\t[Program]'`

  The report is fetched based on the action that you specify. For details of `'[Report type] [Tree data wanted] [Program]'` refer to Action Number 6500. See *Customizing EPD.Connect* for details.

# Glossary

This glossary includes general terms and terms specific to EPD.Visualizer used in this manual and in online help.

### 3D Scene

A collection of objects, including graphics, text, or lights, that display in the EPD.Visualizer graphics window.

### Active Coordinate System

The set of axes currently in control.

### Ambient Light

A diffuse background lighting radiating from all directions. Ambient light applies to all planar geometry in the 3D scene. You can set the brightness of the light, or turn the ambient light on or off.

### Back Facing Polygons

A rendering option that controls the display of back facing polygons. When Back Facing Polygons is on, back facing polygons are removed. Turning this option on might increase the rendering speed of your 3D scene. The Back Facing Polygons option is beneficial when objects in your 3D scene are closed polyhedrons. Likewise, open polyhedrons are displayed with back facing polygons missing when viewing through open areas.

### Binary Graphic File

An encoded file containing the definitions of geometric objects and colors in EPD.Visualizer binary format. This file cannot be edited or viewed as readable text and is generated by software on your host CAD system.

Binary files usually are assigned a `.gbf` extension. These files are extremely compact and can be processed many times faster than Textual Graphic Files.

### Buffering

See “Double Buffering” on page Glosssary-2 and “Single Buffering” on page Glosssary-6.

### Click

Press and release a mouse button.

### Context-Sensitive Help

Access online help by selecting a specific menu name, button, tool, or slider scale.

### Database Management System

A computer program for general-purpose data storage and retrieval.

### DBMS

See Appendix A, “Database Interface”.

### Directional Light

A directional light has its source at infinity and shines through the origin of your 3D scene. All light rays from a directional light are parallel and uniformly illuminate all objects in your 3D scene, regardless of location.

### Double Buffering

A rendering option that does not display the images on the graphics screen until after they have been completely rendered. This dramatically reduces screen flickering.

### Double Click

Quickly press and release the SELECT mouse button twice.

### Drag

Hold down the SELECT mouse button and move the pointer across the screen.

File Selection Box

A special dialog box with controls for opening directories and selecting and opening files. You can open directories by double clicking on the directory name or by selecting the directory name and clicking the Filter button. The selected file appears in a text box below the directory list. You can open a file by double clicking on the file name in the file list.

Flat Shading

A rendering option that indicates all polygon faces are drawn as flat surfaces.

Geometry Units

The unit type specified in a graphic file. This unit type is applied to all geometry specified in the file.

Global Axes

The axes whose orientation is always represented in the lower-left corner of the Graphics Window, also called Project Axes. The origin of these axes is at the point X0 Y0 Z0 of the 3D scene. The specifications of local axes are relative to these axes.

Gpath

A path that has a referential associated with each vertex. The term comes from “generalized path .” A Gpath can have a single vertex.

Hidden Line

A rendering option that displays a solid view of the 3D scene with black polygon edges and lights off.

Item

Geometry (text, lights, paths, and objects) that is part of your 3D scene in EPD.Visualizer.

Local Axes

The local axes coordinate system is attached to the items on the selection list and rotates with the items. The initial default condition of local axes is parallel to the project axes with their local origin at the geometric center of the selection list. If the Use Local CPL button is selected in the Translate/Rotate dialog box, the local axes have a different condition explained in “Translate/Rotate” on page 3-22.

### Matrix Duplication

A lighting feature that enables you to copy a selected light into a matrix of evenly spaced rows and columns. This option works with all light types except directional.

### Menu Bar

The bar that displays across the top of the EPD.Visualizer window containing the menu names.

### Metafile

An ASCII file that contains a complete description of a EPD.Visualizer session. This includes all geometric objects, views, view sequences, lights, paths, color definitions, annotation, and control settings, such as working units and slider scale settings.

### Mirror Reflectance

A surface materials option that determines how much light a material reflects. Wood, for example, is not reflective, whereas chrome is highly reflective.

### Movie

A sequence of image files that are displayed at a specific speed to simulate animation.

### Movie File

An ASCII file that contains information about a movie. The information includes image type, and start and end frame number.

### Object

The design geometry that you import to EPD.Visualizer from your CAD system.

### Path

A sequence of line segments consisting of any number of points, along which you can traverse the camera or objects.

### Point

Move the pointer until it rests on the item of choice.

Point Light

A point light is like a tiny light bulb. It occupies one point and radiates light equally in all directions in your 3D scene.

Primary View

A view that you can return to if you become disoriented while moving through your 3D scene. The initial primary view is defined when you add your first graphics file. The primary view is saved with a metafile and is later restored when you open a metafile.

Project Axes

The axes whose orientation is always represented in the lower-left corner of the Graphics Window, also called Global Axes. The origin of these axes is at the point X0 Y0 Z0 of the 3D scene. The specifications of local axes are relative to these axes.

Radiosity

An advanced rendering method that takes into consideration radiated light from diffusely reflecting surfaces in a 3D scene. Radiosity can be applied only to small 3D scenes.

Ray Tracing

A means of displaying reflections and refractions in a 3D scene. Ray tracing tracks beams of light from the target back to the light source, taking into account reflections, mirroring, transparency, shadows, and refraction. Ray tracing results in a transient pixel image that must be saved as a raster image.

Reference

The combination of name, orientation, and position information that describes a vertex in a Gpath.

Referential

The combination of orientation and position information that describes an object, plane, or reference.

Rendering

The process of displaying a 3D scene in the EPD.Visualizer graphics window. EPD.Visualizer renders each 3D scene into an image file that contains color information for each pixel.

### Shell Polygon

A network of interconnected polygons that share common points and result in a unit typically called a polyhedron. Shell polygons can be smooth shaded.

### Selection List

A selection list stores objects that you select. After the objects are added to the selection list, you can Hide, Show, Translate, Rotate, Delete, or Traverse them. You can build a selection list with the Select and Select By Volume tools and the Build Selection List dialog box.

### Selection Set

A named collection of objects that can be saved and later added to the selection list.

### Single Buffering

A rendering option that displays the images on the graphics screen as they are being rendered.

### Slider Scale

A control for setting numeric values by dragging the slider or clicking on the bar. You can customize many of the EPD.Visualizer slider scales by clicking the MENU mouse button over the scale value.

### Smooth Shading

A rendering option that displays polygons so you do not see the edges between them. This option gives the objects a smooth appearance. These polygons are defined as shells in the graphic files and are denoted with an `#SP` in the textual graphic file.

### Specular Glare

A surface materials option that determines the focus of the beam of light reflected by a material. High specular glare produces a small, focused shiny spot; low specular glare produces broader and more scattered shiny spots.

### Specular Hue

A surface materials option that determines whether the specular reflection of a material displays in color or in shades of gray.

### Specular Intensity

A surface materials option that determines the sharpness of a reflection. Dull surfaces typically appear matted or flat. Shiny surfaces typically appear hard and polished.

### Spot Light

A type of point light that can be focused on a particular target area. Spot lights can be sharp edged, fuzzy edged, or variable focused.

### SQL

The ANSI standard language used to manipulate information in a relational database. SQL is the language used in Oracle relational database management systems.

### Textual Graphic File

A text file containing the definitions of geometric objects, colors, and paths in EPD.Visualizer ASCII format. You can edit this file with a standard text editor. The Textual Graphic file is generated by software on your host CAD system and is usually assigned a `.gaf` extension.

### Text Properties

Specific settings for text, consisting of font, font type (variable or fixed), text alignment, font size, text color, and character spacing.

### View

Specific settings with a specific camera orientation, including camera position, camera target, and perspective angle. You can save, label, and retrieve individual views for display at a later time. You can save any number of views. Views are saved with metafiles.

### View Sequence

A sequence of saved views that you play back interactively or by time delay. You can define any number of view sequences.

### Wire Frame

A rendering option that displays all surface edges of a 3D scene as lines.

### Working Units

The unit type (such as feet, miles, or meters) used for calculations when you perform walk-through operations, object traversal, or item verification. You define the working units by selecting Options > Preferences.

# Index

## Symbols

.gaf files B-1

## A

Ambient light
  description 2-6
Angular movement 1-9
Assembly data trees F-10
Assembly design data
  storing F-2
Axes
  global 3-22, 3-23
  local 2-14, 3-22
  project 2-14
  verifying 3-25

## B

Back facing polygons
  description 3-47
Binary graphic file
  adding 3-5
  color definition C-6
  control options C-3
  conversion to textual file D-1
  converting to a textual file 1-7
  description 1-7, C-1
  example C-15
  extents definition C-5
  format C-2
  geometry specifications C-11
  object definition C-8
  removing 3-7
Box zone
  creating 4-4
  purpose 4-4
Buffering
  display 3-47
  double 3-47
  single 3-47

## C

CADDS interface
  assembly data trees F-10
  converting entities to explicit F-6
  converting tessellation data F-8
  creating tessellation data F-2
  errors F-6
  example F-5
  generating _td files F-10
  messages F-6
  multiple input and output F-9
  purpose F-1, F-2
  restrictions F-6
  updating the _td file F-6
  using F-3
Camera
  measuring distance 9-8
  setting position 9-6
  setting position and target 9-7
  setting target 9-7
Camera traversal
  color blending 3-70

focus 3-67
overview 2-19
purpose 3-66
starting 3-69
view direction 3-69
Centering
quick view centering 1-28
Clipping
description 7-5
starting 7-5
Color
changing in graphics window 3-74
changing item color 2-17
defining 1-19
defining for sections 7-8
deleting 1-20
enhancing item color 2-17
selecting 1-20
verifying 1-20
Color blending
description 3-70
using for a movie 3-70
Color editor
defining plane color 7-4
defining section color 7-8
purpose 1-19
Coordinate system
active 3-23
Cartesian 1-8
local 2-14, 3-22, 3-23
xyz 3-22
CVDPA_CONFIG_DIR environment variable H-2
CVDPA_SECTIONS_DIR environment variable 7-30, H-10
CVDPA_SPACING environment variable H-11
CVDPA_USE_MIRFLAG environment variable H-6
CVDPA_ZONES_DIR environment variable H-11
CVPATH environment variable H-5
Cylinder zone
creating 4-7
purpose 4-7


# D


Database
creating A-2
disconnecting from 3-86
opening 3-85
purpose 3-84
querying 3-90
querying data 3-93
saving reports 3-94
using selection keys 3-88
Database definition file
example A-3
purpose A-2
structure A-7
Database file
description A-2
Database Management System
See DBMS
DBMS
connecting to 3-84, 3-86, A-1
SQL interface A-11
Deselection
menu operation 2-9
tool 9-12
Dialog boxes
displaying 1-15
purpose 1-15
DIMENSION III interface
ASCII graphic file contents E-6
color numbers E-6
description E-2
error messages E-7
generating graphic files E-3
processing multiple viewfiles E-4
processing single viewfiles E-3
Directional light
defining 3-54
definition attributes B-8
description 2-5, 3-51
Documentation, printing from Portable Document Format (PDF) file xxiii
Dynamic interference checking
purpose 2-21
results 5-9
setting up 5-4


# E


Environment Variables
help 2-23
PVS_CONCAVE_PERFORM 3-6
Environment variables
all variables explained H-1

CVDPA_SECTIONS_DIR 7-30
PVS_PATHS_DIR 3-34
PVS_SECOND_VIEW_SIZE 3-44
PVS_SECTION_AXIS 7-10, 8-9
PVS_TEMP_COLOR 3-45
PVS_UNNAMED_REF B-13
setting for zones 4-3

# F

Filter
using 1-18
Flat shading
description 2-2

# G

Gpath
characteristics 3-28
creating or editing 3-30
making items traverse 3-18
overview 2-19
sectioning from references, configuring 7-17
sectioning from references, performing 7-19
sectioning from references, saving results 7-20
traversal options 3-20
Gpath Multiple Sectioning window 7-17
Graphic (ASCII) file
generating F-9
reducing the size F-6
reorienting objects F-11
Graphic file
adding 1-7, 1-29, 3-5
binary 1-7
sectioned file 7-33
textual 1-7, 7-32
Graphics output
from Section window 7-36
PostScript 3-11
Graphics window
changing color 3-74
description 1-11
overview 1-11

# H

Help
accessing on line 2-23, 3-98
context-sensitive 2-23, 3-98
on basics 3-98
on environment variables 2-23, 3-99
on general information 2-23
on glossary terms 2-23, 3-99
on palettes 9-1
on release notes 3-99
using 3-98
Hidden line
description 2-2
Hiding 2-13, 3-14
HOOPS_PICTURE environment variable H-2

# I

Interfaces
CADDS 1-5, F-1
DIMENSION III 1-5, E-2
MEDUSA 1-5
Interference checking
creating a secondary selection 5-7
description 2-21
dynamic 2-17
item traversal overview 2-20
purpose 3-96, 5-2
reporting the results 5-10
reviewing the results 5-9
setting up your session 5-4
starting 5-3
using a secondary selection 3-82
using with sectioning 7-11
Items
definition of 1-6
deleting 2-13
how to rotate 3-23
how to translate 3-23
overview of moving 2-14
overview of rotating 2-14
traversing paths and Gpaths 3-18

# L

Lights
  defining 3-51, 3-53
  deleting 3-59
  editing 3-57
  kinds 3-51
  managing 2-5
  setting brightness 2-6
Linear movement 1-9
Lists
  scrolling 1-15
  selecting multiple entries 1-16
Local Axes Verification dialog box 3-25
Local CPL toggle button 3-22

# M

Main window 1-11
Material definitions
  attributes B-10
  defining B-10
Material properties
  assigning 3-37
  creating 3-36
  defining 2-17
  deleting 3-38
  description 3-35
  editing 3-38
Measuring distance
  between a point and section 7-24
  between an endpoint and section 7-25
  between graphics files 2-12
  between items 9-8
  between objects 2-12
  between two endpoints 7-23
  between two points 7-21
  between two sections 7-26
  considerations 7-27
  from camera to item 9-8
  from camera to object 2-11
  in sections 2-12, 7-20
Menu
  bar description 1-14
  displaying items 1-14
  using 1-14
Metafile
  description 1-7, 3-4
  opening 1-29, 3-4
  saving 3-8
  saving primary view 3-41
  saving selection sets 2-10, 3-79, 3-80
  saving views 3-60
Model axis motions
  behavior 1-26
  example 1-27
  purpose 1-26
Mouse
  changing dynamics of 3-74
  mouse viewing 9-9
  permanent dynamics 1-21
  section view behavior 7-8
Move Palette
  purpose 1-12
Move to Coordinates dialog box 3-27
Movie
  loading G-4
  playing G-6
Movie recorder
  color blending 3-70
  defining 3-71
  description 3-70

# N

New session
  starting 3-3
Normals
  displaying 3-78
  orientation of references 3-32

# O

Object
  definition of 1-6
Object definition
  discipline attribute B-14
  example B-18
  format B-14
  group attribute B-14
Object Path Traversal dialog box 3-19
Object Path Traversal Options dialog box 3-19
Object tagging
  automatic 3-95, 9-13, A-16
  manual 3-95, 9-13

purpose 9-13
OLD_GLOBAL_SPELLING environment variable H-6
Orientation
constant 2-16, 3-20
to path 2-16, 3-20
vertical 2-16, 3-20

# P

Palettes
hiding or showing 1-11, 3-73
kinds 1-11
Move Palette 1-12
moving left or right 1-11, 3-73
stretching or shrinking 1-11
Tool Palette 1-12, 9-2
Turn/Orbit Palette 1-12, 9-19
walk-through 9-15
Part data
storing F-2
Path
creating or editing 3-30
making items traverse 3-18
overview 2-19
purpose B-12
traversal options 3-20
Path editor
purpose 3-28
Path Editor dialog box 3-30
Path Router dialog box 3-31
Permanent Zoom tool 9-5
Perspective angle
description 2-3
Perspective view
changing 2-3
Plane
clipping the scene 7-5
constant orientation 8-4
cutting plane 7-5
defining by digitizing 6-6
displaying 7-3
moving along a path 8-3
moving to a Reference 8-9
moving to the selection list 8-10
moving with objects selected 8-6
orienting 6-6
positioning 6-5
rotating 8-8
sectioning 7-5
sectioning the scene 7-6
sizing 7-3
to path orientation 8-5
translating 8-7
vertical orientation 8-4
viewing 7-5
Plane dynamics
setting 8-3
Point light
defining 3-55
definition attributes B-9
description 2-5, 3-51
Polygons
back facing 3-47, F-6
flat B-15, B-16, C-11
right-hand rule B-14
shell B-15, B-16, C-11, C-12
shell face B-17
shell hole B-17
shell points B-16
Polyhedron
definition C-11
Polyline B-17, C-14
PostScript graphics output 3-11
Primary view
description 2-7, 3-41
setting 2-7
Printing documentation from Portable Document Format (PDF) file xxiii
Prompt window 1-11
PV_SYSTEM environment variable H-4
PVS_CADDS_COLORS environment variable H-5
PVS_COLOR_FILE environment variable H-7
PVS_CONCAVE_PERFORM environment variable 3-6, H-5
PVS_EXTERNAL environment variable H-6
PVS_LOCAL_CPL environment variable H-8
PVS_NORMAL_AXIS environment variable 3-26, H-8
PVS_ORTHO_MODE environment variable H-3
PVS_PATHS_DIR environment variable 3-34
PVS_PATHS_DIRPVS_PATHS_DIR environment variable H-8
PVS_PRINT_COMMAND environment variable H-9
PVS_PRINT_FACTOR environment variable H-9

PVS_SECOND_VIEW_SIZE environment variable 3-44
PVS_SECTION_AXIS environment variable 7-10, 8-9, H-10
PVS_STEREO environment variable H-3
PVS_STEREO_ANGLE environment variable H-3
PVS_TEMP_COLOR environment variable 3-45, H-3
PVS_UNNAMED_REF environment variable B-13, H-8

# Q

Query type
  defining 3-87
  purpose 3-86

# R

Radiosity
  description 2-2
  setting controls for 3-50
Raster image
  capturing 3-9
Ray tracing
  setting controls for 3-49
References
  characteristics 3-29
  defining 3-32
  in sections, information about 7-10
  sectioning from multiple 7-14
Rendering
  controlling 1-12, 9-15
  methods 2-2
  ray tracing 2-3
  suppressing 1-9, 1-12, 9-15
Reports
  of interference checking results 5-10
  saving database reports 3-94
Reset Orientation button 3-27
Reset Position button 3-27
Rotation
  how to rotate selections 3-23
  interference checking 2-17
  overview 2-16
  purpose 3-22
  restoring position 3-27

# S

Save Path dialog box 3-34
Scene
  clipping 7-5
  sectioning 7-6
Screen axis motions
  behavior 1-22
  example 1-23
  purpose 1-21
Secondary selection
  clearing 3-83
  creating 3-82
  creating for interference checking 5-7
Secondary Views 3-44
Section
  displaying in a separate view 7-6
  measuring distances 7-20
  saving in a file 7-29
Section window
  graphics output 7-36
  mouse behavior 7-8
  view options 7-9
Sectioning
  from multiple references, configuring 7-17
  from multiple references, performing 7-19
  from multiple references, saving result 7-20
  purpose 3-97, 6-2
  starting 6-3
Sectioning Motion dialog box 8-2
Selectability mask
  setting 3-82
Selection
  copying to secondary selection 3-82
Selection list
  adding to 2-9, 3-82
  building 2-9, 2-10, 3-79, 3-81
  clearing 2-9
  purpose 2-9
  rotating 3-24
Selection mask
  purpose 2-9, 3-79
  setting 2-9
Selection set
  creating 3-79
  defining 2-10, 3-80
  description 3-80
  editing name 3-80
  highlighting 3-81

purpose 2-10
Session
obtaining information about 1-30, 3-12
Shadows
setting controls for 3-48
Showing 2-13, 3-14
Slider scales
Color Composition 1-19
customizing 9-22
location of 1-11, 1-13
Move Distance 9-18
Turn Angle 3-20
Turn/Orbit Angle 9-22
Smooth shading
description 2-2
Spot light
defining 3-56
definition attributes B-9
description 2-5, 3-52
SQL interface
environment setup A-16
purpose A-1, A-11
required tables A-11
SQL query statement
creating 3-90
selecting from DBMS 3-92
selecting from file 3-91
Static interference checking
purpose 2-21
Status window 1-11
Substitution variable
defining 3-89
description 3-90
editing 3-89
purpose 3-88
Surfaces
assigning material properties 3-35

# T

Tags
attaching to objects 2-22
Target
choosing 2-15
Tessellation data
converting F-8
creating F-2
Text
attaching to objects 2-22
changing properties 9-13
creating 2-22
defining properties 2-22
deleting 2-22
editing 2-22, 9-12
entering 9-12
tagging with 2-15
Text properties
changing 9-13
defining 2-22
setting 3-39
Textual graphic file
adding 3-5
color definition B-7
contents B-2
control options B-4
conversion to binary file D-1
converting to a binary file 1-7
DBMS control options B-5
description 1-7
example C-15
extents definition B-6
format B-3
Gpath definitions B-13
Gpath multiple sectioning B-19
light definitions B-8
material definitions B-10
object definitions B-14
path definitions B-12
removing 3-7
size limitation B-2
Tool Palette
description 1-12, 9-2
Translate/Rotate Selection List dialog box 3-23
Translation
how to translate selections 3-23
interference checking 2-17
overview 2-15
purpose 3-22
restoring position 3-27
text 2-15
Traversal
camera traversal overview 2-19
camera traversal, performing 3-66
Options 3-20
overview 2-19
paths contrasted to Gpaths 3-16

purpose 2-16
traversing items on Gpaths 3-18
traversing items on paths 3-18
Tree Select and Scroll tool 9-5
Turn/Orbit Palette
description 9-19
purpose 1-12


# V


Variables, environment, help on 2-23
Verification information
obtaining A-14
Verifying a point 7-28
View
centering 1-28
changing 3-43
concept of viewing 1-9
defining 3-61
deleting 3-61
displaying all 3-42
displaying primary view 3-41
editing view name 3-61
managing 3-60
perspective 3-45
perspective angle 2-3
Primary view 2-7
rendering 3-46
saving 2-7
separate view of section 7-6
setting 2-7
View sequence
adding to 3-63
defining 2-7, 3-62
deleting 3-63
displaying interactive playback 3-64
displaying time delay playback 3-65
editing sequence names 3-63
playback 2-7
rearranging 3-63
Visibility
controlling 2-13
Visualizer
features 1-3
overview 1-6
querying 2-11
quitting 1-30, 3-13
starting 1-29
Visualizer movie player
executing G-2
quitting G-6
Visualizer window
customizing 1-11
parts of 1-11
VS_CLIENT_FILE environment variable H-6


# W


Walk-through palette
purpose 9-15
Wire frame
description 2-2
Working units
changing 3-74
defining 2-11


# X


XIDPTR (Extended Identification Pointer)
description A-8
example A-8
verifying A-9


# Y


YMIRFLAG environment variable H-6


# Z


Zones
creating 4-2
description 4-2
purpose 4-2
setting environment variables for 4-3
storing 4-3